Promotion of Open Science in Requirements Engineering – Leveraging the Open Research Knowledge Graph for FAIR Scientific Information

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

June 24th, 2024, Reykjavik, Iceland

Welcome!

Oliver Karras

Alessio Ferrari

Davide Fucci

Davide Dell’Anna

Schedule and Table of Content

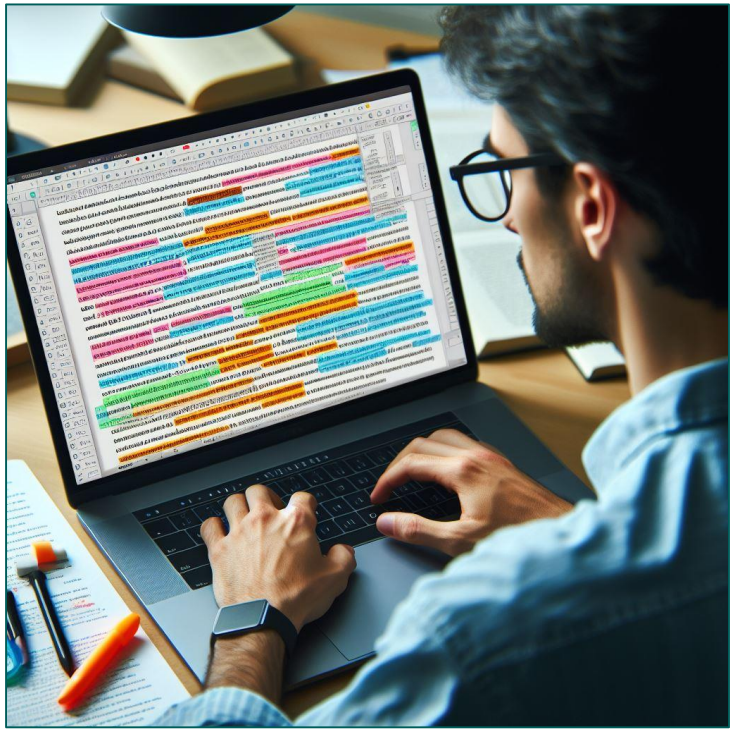
Session	Time	Table of Content	Style	Speaker
Theoretical	09:00 - 09:25	1. Welcome (5 min)	Presentation	All organizers
		2. Introduction to open science in RE (10 min)	Presentation	Alessio Ferrari
		3. Introduction to the ORKG (10 min)	Presentation	Oliver Karras
Practical	09:25 - 10:15	4. Create a FAIR-annotated publication for the ORKG (50 min)	Exercise	Oliver Karras
		4.1 Set up an Overleaf project for an exemplary publication	Sub-exercise	All organizers
		4.2 Use the LaTex package SciKGTeX to annotate the publication	Sub-exercise	All organizers
		4.3 Generate PDF with embedded FAIR scientific information	Sub-exercise	All organizers
		4.4 Optional: Upload the FAIR-annotated publication to the ORKG	Sub-exercise	All organizers
Break	10:15 - 10:45	Coffee break		
Practical	10:45 - 11:45	5. Use the ORKG based on a RE use case (60 min)	Exercise	Oliver Karras
		5.1 Add an exemplary publication to the ORKG	Sub-exercise	All organizers
		5.2 Describe the scientific information of the publication in the ORKG	Sub-exercise	All organizers
		5.3 Create an ORKG comparison of the publications added by participants	Sub-exercise	All organizers
		5.4 Publish the created ORKG comparison as a citable digital artifact	Sub-exercise	All organizers
		5.5 Optional: Create visualizations for the created ORKG comparison	Sub-exercise	All organizers
		5.6 Optional: Retrieve the information with the SPARQL endpoint	Sub-exercise	All organizers
Feedback	11:45 - 12:15	6. Reflection of the tutorial with the participants (25 min)	Discussion	All organizers
		7. Farewell and closing (5 min)	Presentation	All organizers

Teaser: REFSQ'25 Open Science Competition

Challenge 1:

Annotate your REFSQ'25 paper with SciKGTeX.

The accepted paper, best annotated with SciKGTeX, will be awarded the **Best ORKG Annotation Award** (prize:100€).



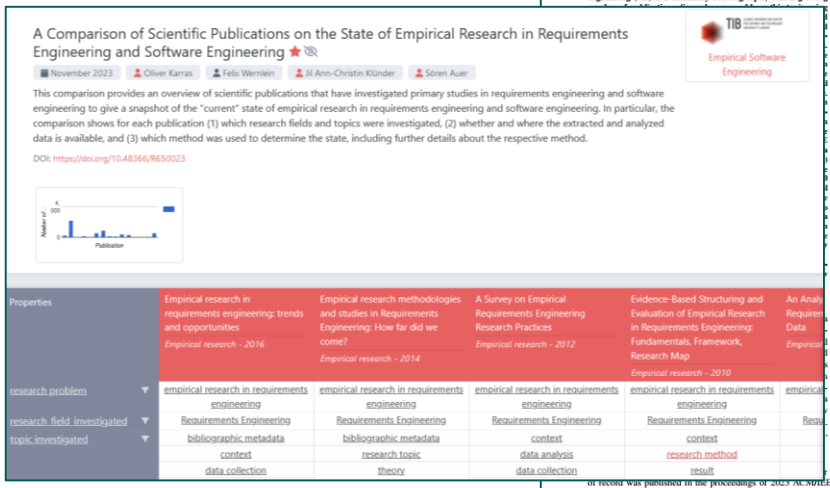
SciKGTeX

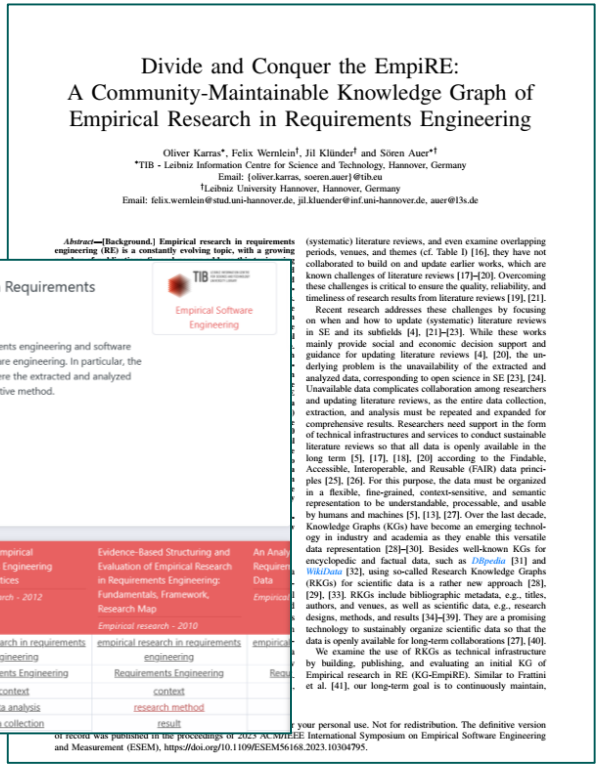
Challenge 2:

Enrich your RESFQ'25 paper with an ORKG comparison.

The accepted paper, enriched with the best ORKG comparison, will be awarded the **Best ORKG Comparison Award** (prize: 200€).

Open Research Knowledge Graph


A Comparison of Scientific Publications on the State of Empirical Research in Requirements Engineering and Software Engineering


Divide and Conquer the EmpiRE:
A Community-Maintainable Knowledge Graph of Empirical Research in Requirements Engineering

TIB LEIBNIZ INFORMATION CENTRE FOR SCIENCE AND TECHNOLOGY UNIVERSITY LIBRARY

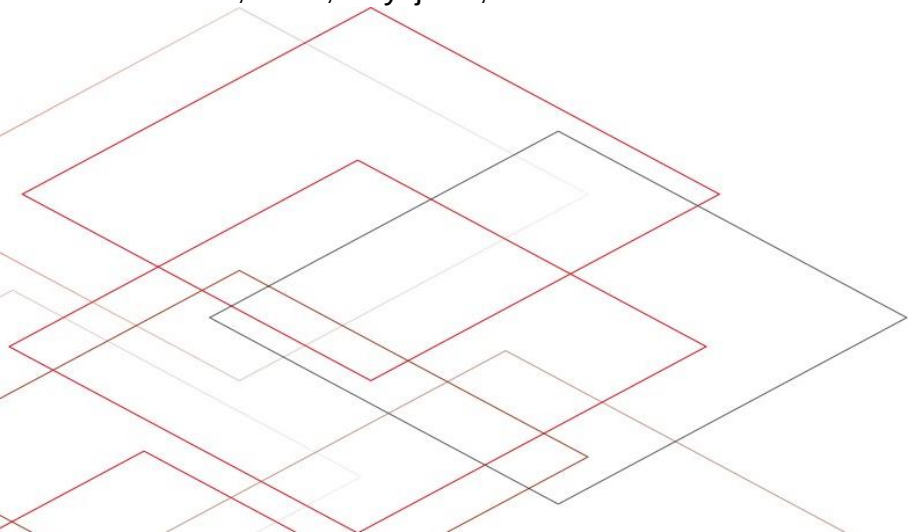
Introduction: Open Science and FAIR in Software Engineering and Requirements Engineering

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

June 24th, 2024, Reykjavik, Iceland



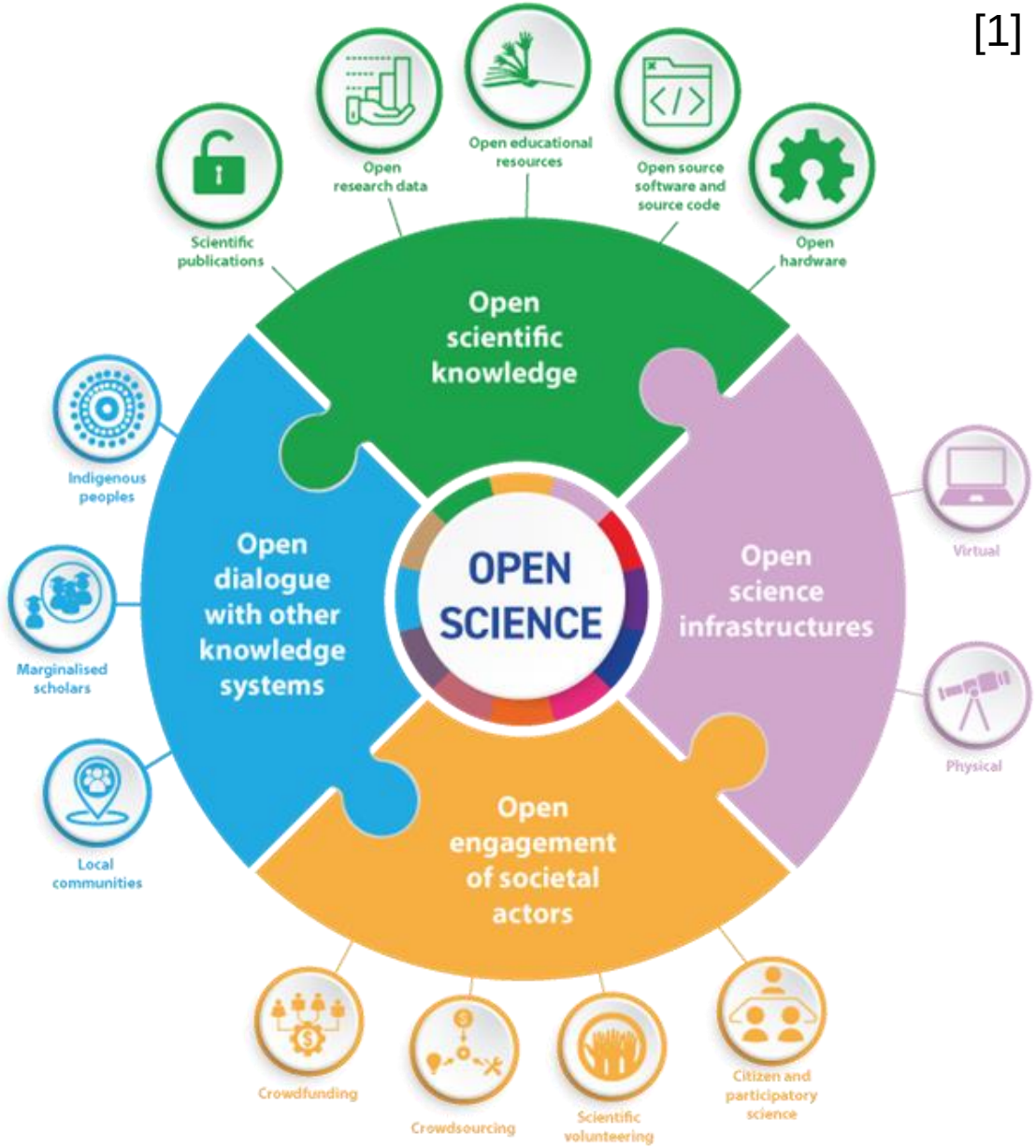
Leibniz Association

What is Open Science?

Open science is the movement to make scientific research (including publications, data, physical samples, and software) and its dissemination **accessible** to all levels of society, amateur or professional.

- Open methodology
- Open source
- Open data
- Open access
- Open peer review (identities, reports, community participation)
- Open educational resources

[1]



[1] UNESCO: *UNESCO Recommendation on Open Science.* DOI: 10.54677/MNMH8546, 2021.

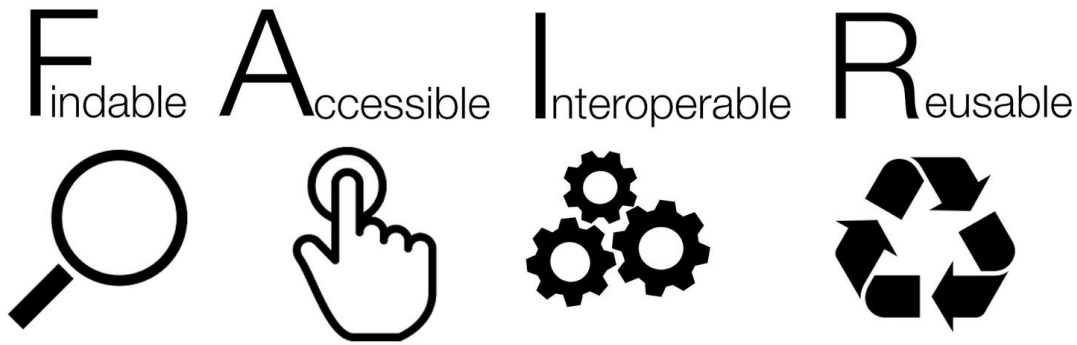
What are the FAIR Principles?

Scientists increasingly rely on computational support to deal with increasing amounts of data.

Focus on **Digital** Assets (Data & Knowledge)

FAIR Principles[2]:

- **F**indability: Rich metadata, reachable by humans and machines, unique identifiers (DOI)
- **A**ccessibility: Open protocols for data access
- **I**nteroperability: Integration with other data
- **R**euse: Replicable experiments, understandable data (through metadata)

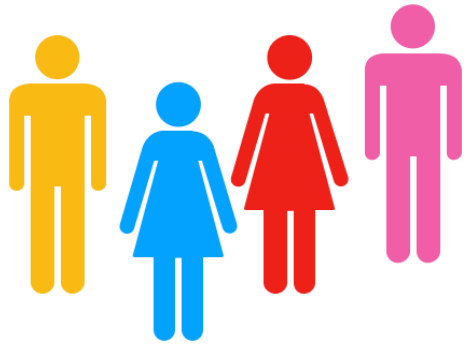


[2] Wilkinson et al.: *The FAIR Guiding Principles for scientific data management and stewardship.* Nature Scientific Data 3, DOI: 10.1038/sdata.2016.18, 2016.

Why Do We Need Open Science and FAIR?

1. **Transparency and Reproducibility:**
   - Open science helps verify the validity of research results
   - Scientists can reproduce experiments, which is crucial for the **credibility**

2. **Collaboration and Innovation:**
   - Collaboration among scientists, and **interdisciplinary** communication
   - Accelerate discoveries and novel **ideas**



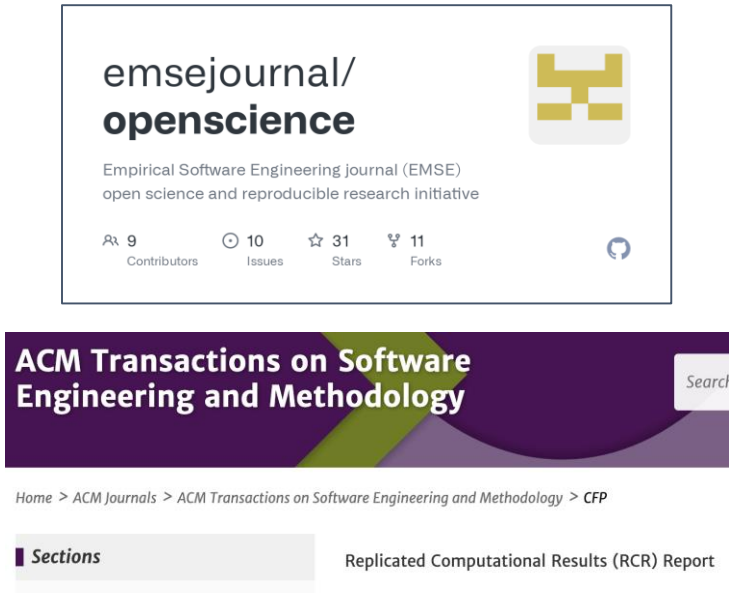
3. **Equity and Inclusion:**
   - Democratize access to knowledge, and enrich viewpoints through **global access**
   - Science accessible to the general public can increase scientific literacy, leading to an **informed society**

Current Initiatives in Software Engineering

Emphasis on Data, **Tools**, Research Results

The ACM badge system (2010)

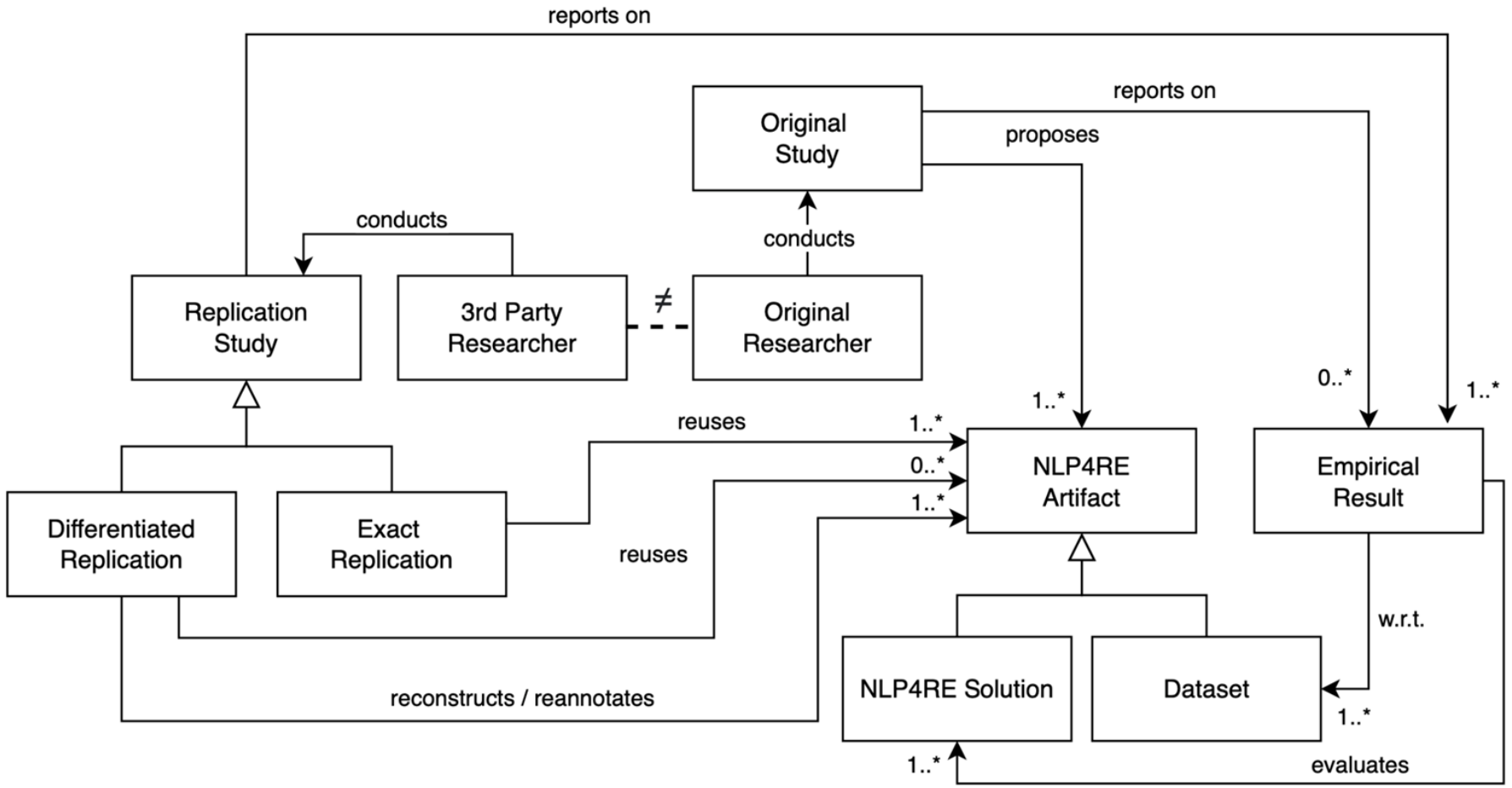
Artifact Evaluation Tracks



Journal Initiatives

Replication and Reuse: Terminology

[3]



[3] Abualhaija et al.: *Replication in Requirements Engineering: The NLP for RE Case.* ACM Transactions on Software Engineering and Methodology, DOI: 10.1145/3658669, 2024.

Example Problems in RE: Replication and Tool Reconstruction

ID	Description	
Rec1	The reconstruction-relevant information and implementation details of the original approach can be ambiguous, imprecise, and incomplete.	**Poor information**
Rec2	If a tool was partially or fully developed and/or evaluated using proprietary data, then there is no guarantee that the reconstructed tool would be identical to the original one since the used data cannot be accessed for reconstruction purposes.	**Proprietary data**
Rec3	Communication with the original authors is not always useful since the actual information sources may not be available anymore.	**Communication**
Rec4	The continuous evolution of the NLP ecosystem entails that some libraries become outdated, unavailable, or not maintained anymore.	**Updates and maintenance**
Rec5	Tools are typically developed as prototypes and not maintained in the long term.	
Rec6	Tool reconstruction is not (yet) valued as a self-standing research contribution, and hence researchers are discouraged to replicate tools overtime.	**Poor value**

Example Problems in RE: Dataset Annotation

ID	Description
Ann1	Some RE-specific categorization tasks lack solid theories that can guide the annotation process.
Ann2	Besides annotation experience and theoretical knowledge, the lack of domain knowledge can limit the accuracy of the annotations.
Ann3	The annotation activity is time consuming due to factors such as language barriers, different individuals' background, and fatigue.
Ann4	The annotation protocol can evolve and thereby necessitate the re-annotation of the data which might, again, cause additional time and effort.
Ann5	Theoretical and practical training resources and opportunities are limited and not adequate for training novice annotators who are often trained during the annotation task by more experienced annotators.
Ann6	The lack of benchmarks entails that annotated datasets enabling comparison against the state-of-the-art are scarce.
Ann7	Available imbalanced datasets pose the challenge of both understanding the minority class and consequently the annotation of new examples thereof.
Ann8	Determining the right amount of context to be shared alongside the annotation raw data is essential and can significantly affect the annotation results.
Ann9	Motivating the annotators poses another challenge since an immediate observation of the impact of a given annotation task is not always possible.
Ann10	Annotators are often not experienced in managing the social aspects or resolving conflicts originating from power, authority, or other social relations.

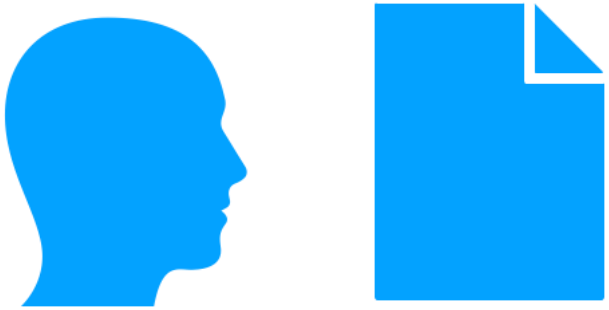
Domain Knowledge

Guidelines

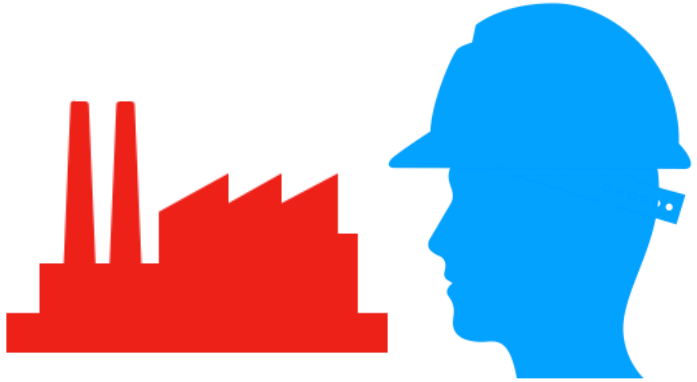
Data Quality

Lack of Benchmarks

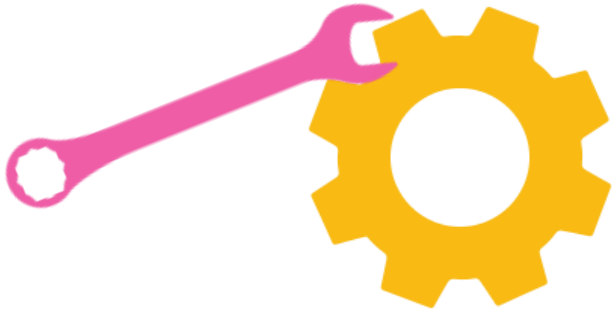
What do We Need?



Clear papers, Clear Communication



Involvement of companies and experts



Tool maintenance

Reward for open science and replication

Thanks to…

Sallam Abualhaija

Fatma Basak Aydemir

Fabiano Dalpiaz

Davide Dell’Anna

Xavier Franch

Davide Fucci

Introduction:
Open Research Knowledge Graph

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

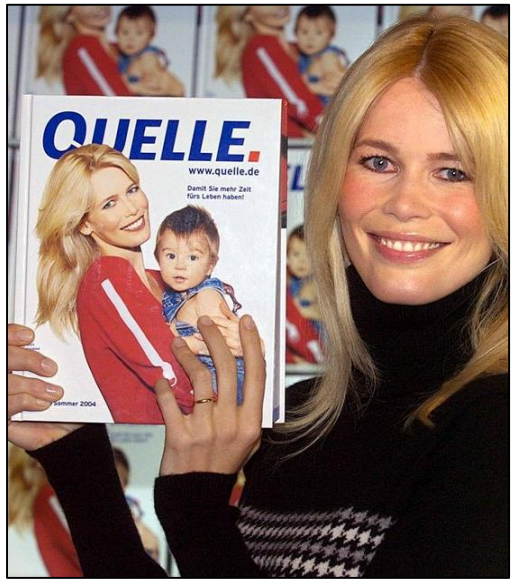
oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

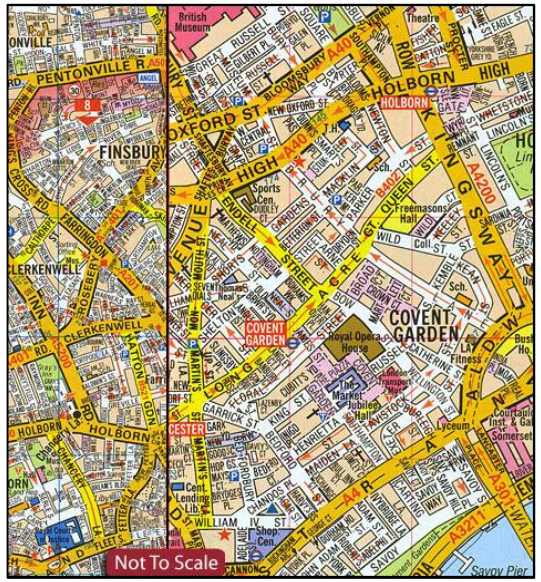
June 24th, 2024, Reykjavik, Iceland

Once Upon a Time, we Communicated with Paper…

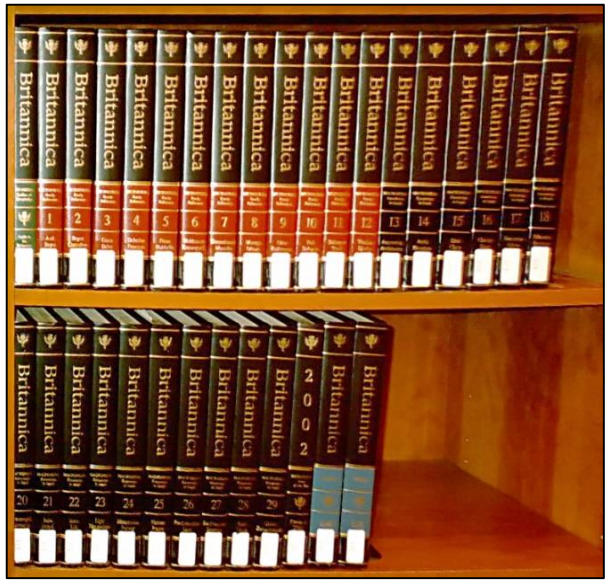
Who still remembers?



Mail order catalogs



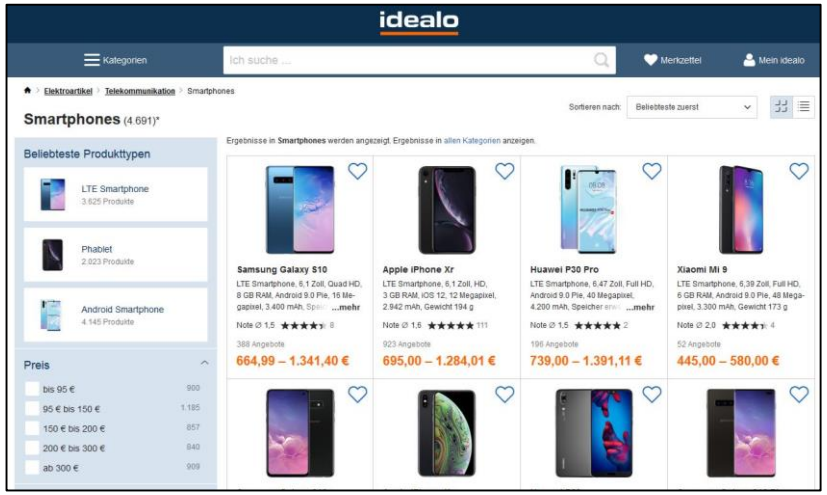
Maps

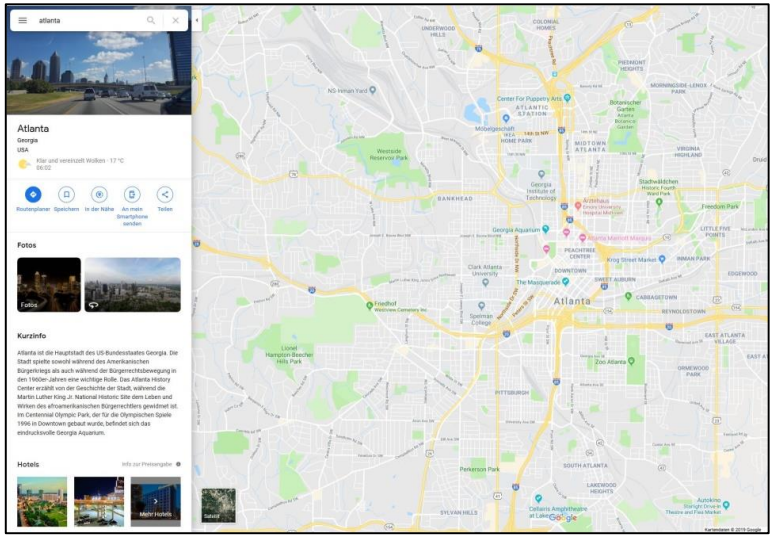


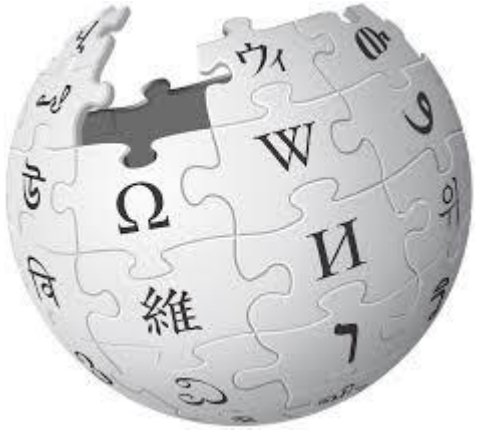
Encyclopedia

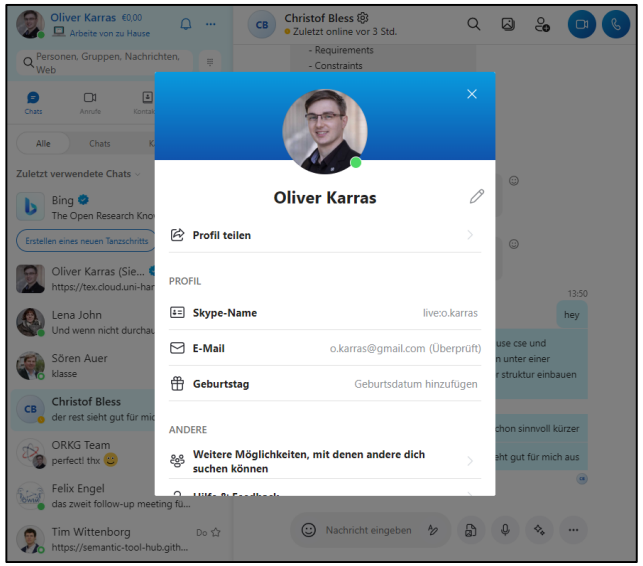
Phone books

… until Digital Transformation (Digitalization)!









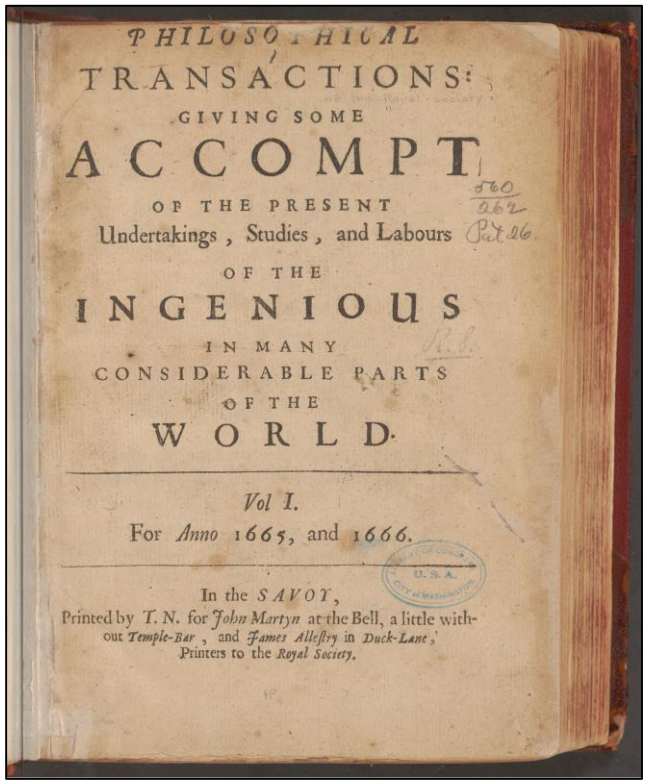
The World of Publishing & Communication has Profundely Changed!

- **New means adapted to the new possibilities**, e.g., platforms
- Completely new **business models**
- More focus on **data, interlinking, services,** and **search**
- **Integration**, **crowdsourcing, and data curation** are important

What has happened in academia in terms of scholarly publishing & communication?

Let's Take a Look

17th century

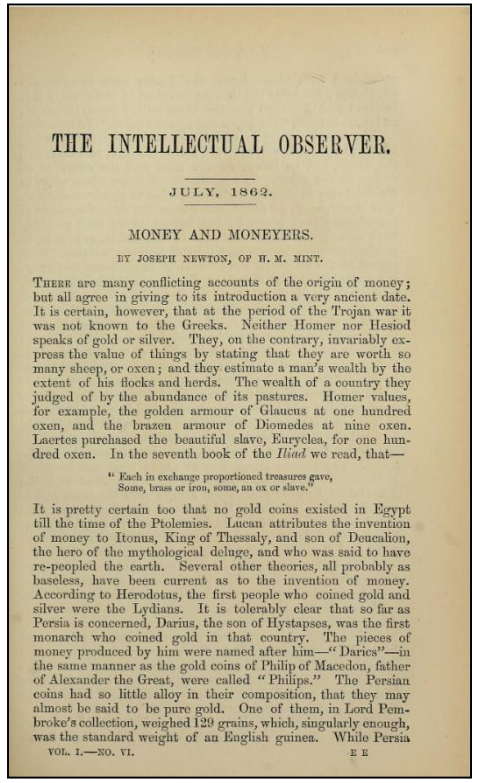


PHILOSOPHICAL
TRANSACTIONS:
GIVING SOME
ACCOMPT
OF THE PRESENT
Undertakings, Studies, and Labours
OF THE
INGENIOUS
IN MANY
CONSIDERABLE PARTS
OF THE
WORLD.

Vol I.
For *Anno* 1665, and 1666.

In the *SAVOY*,
Printed by *T. N.* for *John Martyn* at the Bell, a little without *Temple-Bar*, and *James Allestry* in *Duck-Lane*,
Printers to the *Royal Society*.

19th century



THE INTELLECTUAL OBSERVER.

JULY, 1862.

MONEY AND MONEYERS.

BY JOSEPH NEWTON, OF H. M. MINT.

THERE are many conflicting accounts of the origin of money; but all agree in giving to its introduction a very ancient date. It is certain, however, that at the period of the Trojan war it was not known to the Greeks. Neither Homer nor Hesiod speaks of gold or silver. They, on the contrary, invariably express the value of things by stating that they are worth so many sheep, or oxen; and they estimate a man's wealth by the extent of his flocks and herds. The wealth of a country they judged of by the abundance of its pastures. Homer values, for example, the golden armour of Glaucus at one hundred oxen, and the brazen armour of Diomedes at nine oxen. Laertes purchased the beautiful slave, Eurycles, for one hundred oxen. In the seventh book of the *Iliad* we read, that—

"Each in exchange proportioned treasures gave,
Some, brass or iron, some, an ox or slave."

It is pretty certain too that no gold coins existed in Egypt till the time of the Ptolemies. Lucan attributes the invention of money to Itonus, King of Thessaly, and son of Deucalion, the hero of the mythological deluge, and who was said to have re-peopled the earth. Several other theories, all probably as baseless, have been current as to the invention of money. According to Herodotus, the first people who coined gold and silver were the Lydians. It is tolerably clear that so far as Persia is concerned, Darius, the son of Hystapses, was the first monarch who coined gold in that country. The pieces of money produced by him were named after him—"Darics"—in the same manner as the gold coins of Philip of Macedon, father of Alexander the Great, were called "Philips." The Persian coins had so little alloy in their composition, that they may almost be said to be pure gold. One of them, in Lord Pembroke's collection, weighed 129 grains, which, singularly enough, was the standard weight of an English guinea. While Persia

20th century

A Relational Model of Data for Large Shared Data Banks

E. F. CODD
IBM Research Laboratory, San Jose, California

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed. Changes in data representation will often be needed as a result of changes in query, update, and report traffic and natural growth in the types of stored information.

Existing noninferential, formatted data systems provide users with tree-structured files or slightly more general network models of the data. In Section 1, inadequacies of these models are discussed. A model based on n-ary relations, a normal form for data base relations, and the concept of a universal data sublanguage are introduced. In Section 2, certain operations on relations (other than logical inference) are discussed and applied to the problems of redundancy and consistency in the user's model.

KEY WORDS AND PHRASES: data bank, data base, data structure, data organization, hierarchies of data, networks of data, relations, derivability, redundancy, consistency, composition, join, retrieval language, predicate calculus, security, data integrity
CR CATEGORIES: 3.70, 3.73, 3.75, 4.20, 4.22, 4.29

1. Relational Model and Normal Form

1.1. INTRODUCTION

This paper is concerned with the application of elementary relation theory to systems which provide shared access to large banks of formatted data. Except for a paper by Childs [1], the principal application of relations to data systems has been to deductive question-answering systems. Levein and Maron [2] provide numerous references to work in this area.

In contrast, the problems treated here are those of *data independence*—the independence of application programs and terminal activities from growth in data types and changes in data representation—and certain kinds of *data inconsistency* which are expected to become troublesome even in nondeductive systems.

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for noninferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a number of confusions, not the least of which is mistaking the derivation of connections for the derivation of relations (see remarks in Section 2 on the "connection trap").

Finally, the relational view permits a clearer evaluation of the scope and logical limitations of present formatted data systems, and also the relative merits (from a logical standpoint) of competing representations of data within a single system. Examples of this clearer perspective are cited in various parts of this paper. Implementations of systems to support the relational model are not discussed.

1.2. DATA DEPENDENCIES IN PRESENT SYSTEMS

The provision of data description tables in recently developed information systems represents a major advance toward the goal of data independence [5, 6, 7]. Such tables facilitate changing certain characteristics of the data representation stored in a data bank. However, the variety of data representation characteristics which can be changed *without logically impairing some application programs* is still quite limited. Further, the model of data with which users interact is still cluttered with representational properties, particularly in regard to the representation of collections of data (as opposed to individual items). Three of the principal kinds of data dependencies which still need to be removed are: ordering dependence, indexing dependence, and access path dependence. In some systems these dependencies are not clearly separable from one another.

1.2.1. *Ordering Dependence.* Elements of data in a data bank may be stored in a variety of ways, some involving no concern for ordering, some permitting each element to participate in one ordering only, others permitting each element to participate in several orderings. Let us consider those existing systems which either require or permit data elements to be stored in at least one total ordering which is closely associated with the hardware-determined ordering of addresses. For example, the records of a file concerning parts might be stored in ascending order by part serial number. Such systems normally permit application programs to assume that the order of presentation of records from such a file is identical to (or is a subordering of) the

21st century

AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data

Ricardo Usbeck[1,2], Axel-Cyrille Ngonga Ngomo[1], Michael Röder[1,2], Daniel Gerber[1], Sandro Athaide Coelho[3], Sören Auer[4], and Andreas Both[2]

[1] University of Leipzig, Germany
[2] R&D, Unister GmbH, Germany
[3] Federal University of Juiz de Fora, Brazil
[4] University of Bonn & Fraunhofer IAIS, Germany
{usbeck,ngonga}@informatik.uni-leipzig.de

Abstract. Over the last decades, several billion Web pages have been made available on the Web. The ongoing transition from the current Web of unstructured data to the Web of Data yet requires scalable and accurate approaches for the extraction of structured data in RDF (Resource Description Framework) from these websites. One of the key steps towards extracting RDF from text is the disambiguation of named entities. While several approaches aim to tackle this problem, they still achieve poor accuracy. We address this drawback by presenting AGDISTIS, a novel knowledge-base-agnostic approach for named entity disambiguation. Our approach combines the Hypertext-Induced Topic Search (HITS) algorithm with label expansion strategies and string similarity measures. Based on this combination, AGDISTIS can efficiently detect the correct URIs for a given set of named entities within an input text. We evaluate our approach on eight different datasets against state-of-the-art named entity disambiguation frameworks. Our results indicate that we outperform the state-of-the-art approach by up to 29% F-measure.

1 Introduction

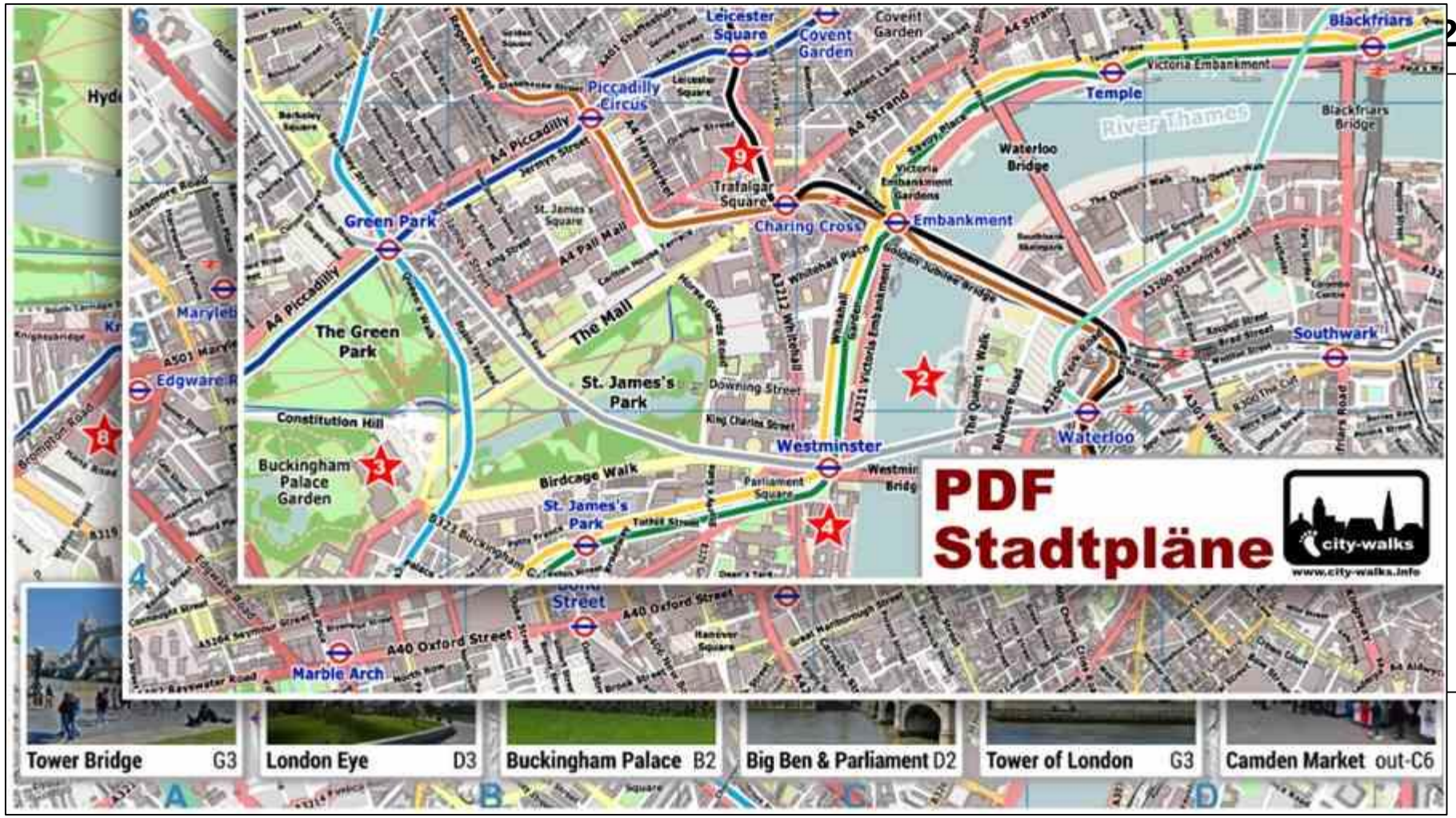
The vision behind the Web of Data is to provide a new machine-readable layer to the Web where the content of Web pages is annotated with structured data (e.g., RDFa [1]). However, the Web in its current form is made up of at least 15 billion Web pages.[1] Most of these websites are unstructured in nature. Realizing the vision of a usable and up-to-date Web of Data thus requires scalable and accurate natural-language-processing approaches that allow extracting RDF from such unstructured data. Three tasks play a central role when extracting RDF from unstructured data: named entity recognition (NER), named entity disambiguation (NED), also known as entity linking [16], and relation extraction (RE). For the first sentence of Example 1, an accurate named entity recognition approach would return the strings `Barack Obama` and `Washington,`

[1] Data gathered from `http://www.worldwidewebsize.com/` on January 4th, 2014.

P. Mika et al. (Eds.) ISWC 2014, Part I, LNCS 8796, pp. 457–471, 2014.

Scholarly publishing & communication has **not changed** (much)!

Let's Take a Look



21st century

AGDISTIS - Graph-Based Disambiguation of Named Entities Using Linked Data

Ricardo Usbeck[1,2], Axel-Cyrille Ngonga Ngomo[1], Michael Röder[1,2], Daniel Gerber[1], Sandro Athaide Coelho[3], Sören Auer[4], and Andreas Both[2]

[1] University of Leipzig, Germany
[2] R&D, Unister GmbH, Germany
[3] Federal University of Juiz de Fora, Brazil
[4] University of Bonn & Fraunhofer IAIS, Germany
{usbeck,ngonga}@informatik.uni-leipzig.de

Abstract. Over the last decades, several billion Web pages have been made available on the Web. The ongoing transition from the current Web of unstructured data to the Web of Data yet requires scalable and accurate approaches for the extraction of structured data in RDF (Resource Description Framework) from these websites. One of the key steps towards extracting RDF from text is the disambiguation of named entities. While several approaches aim to tackle this problem, they still achieve poor accuracy. We address this drawback by presenting AGDISTIS, a novel knowledge-base-agnostic approach for named entity disambiguation. Our approach combines the Hypertext-Induced Topic Search (HITS) algorithm with label expansion strategies and string similarity measures. Based on this combination, AGDISTIS can efficiently detect the correct URIs for a given set of named entities within an input text. We evaluate our approach on eight different datasets against state-of-the-art named entity disambiguation frameworks. Our results indicate that we outperform the state-of-the-art approach by up to 29% F-measure.

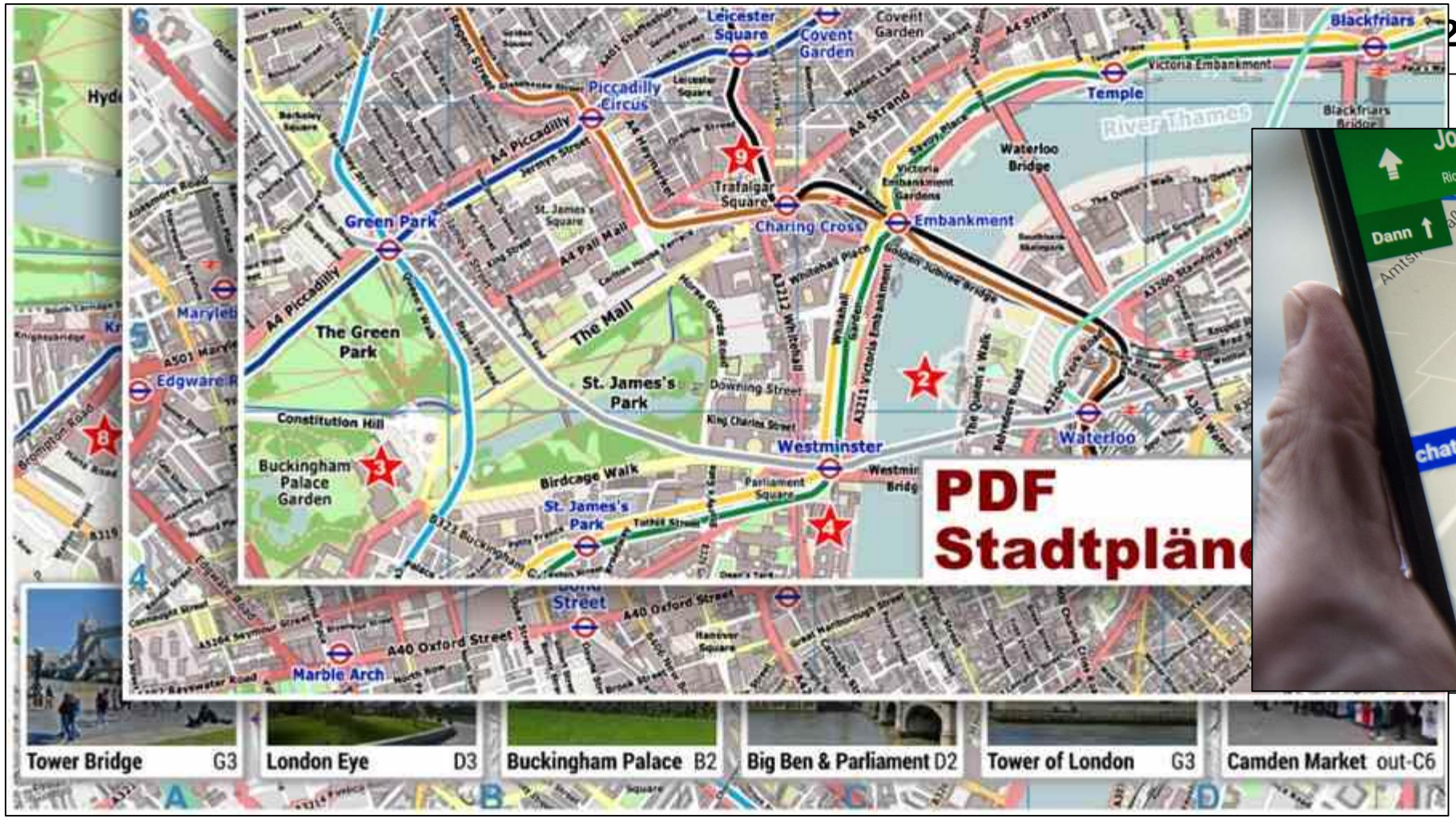
1 Introduction

The vision behind the Web of Data is to provide a new machine-readable layer to the Web where the content of Web pages is annotated with structured data (e.g., RDFa [1]). However, the Web in its current form is made up of at least 15 billion Web pages.[1] Most of these websites are unstructured in nature. Realizing the vision of a usable and up-to-date Web of Data thus requires scalable and accurate natural-language-processing approaches that allow extracting RDF from such unstructured data. Three tasks play a central role when extracting RDF from unstructured data: named entity recognition (NER), named entity disambiguation (NED), also known as entity linking [16], and relation extraction (RE). For the first sentence of Example 1, an accurate named entity recognition approach would return the strings `Barack Obama` and `Washington,`

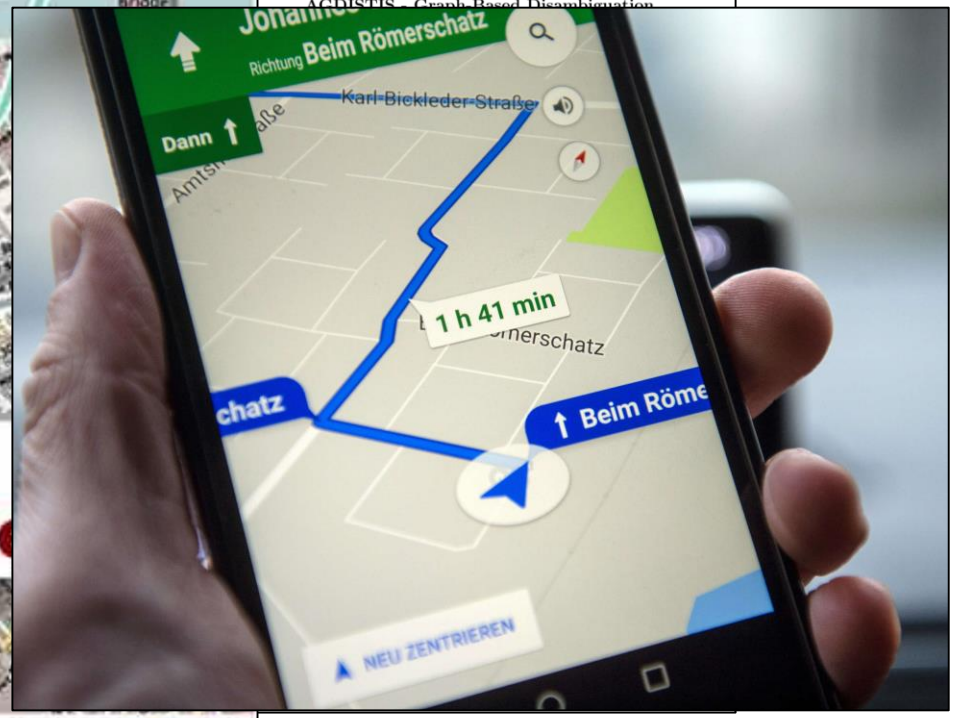
[1] Data gathered from `http://www.worldwidewebsize.com/` on January 4th, 2014.

P. Mika et al. (Eds.) ISWC 2014, Part I, LNCS 8796, pp. 457–471, 2014.

Let's Take a Look



21st century



Rethink How Scientific Knowledge is Communicated

"*The lightbulb was **not** invented by improving the candle.*"

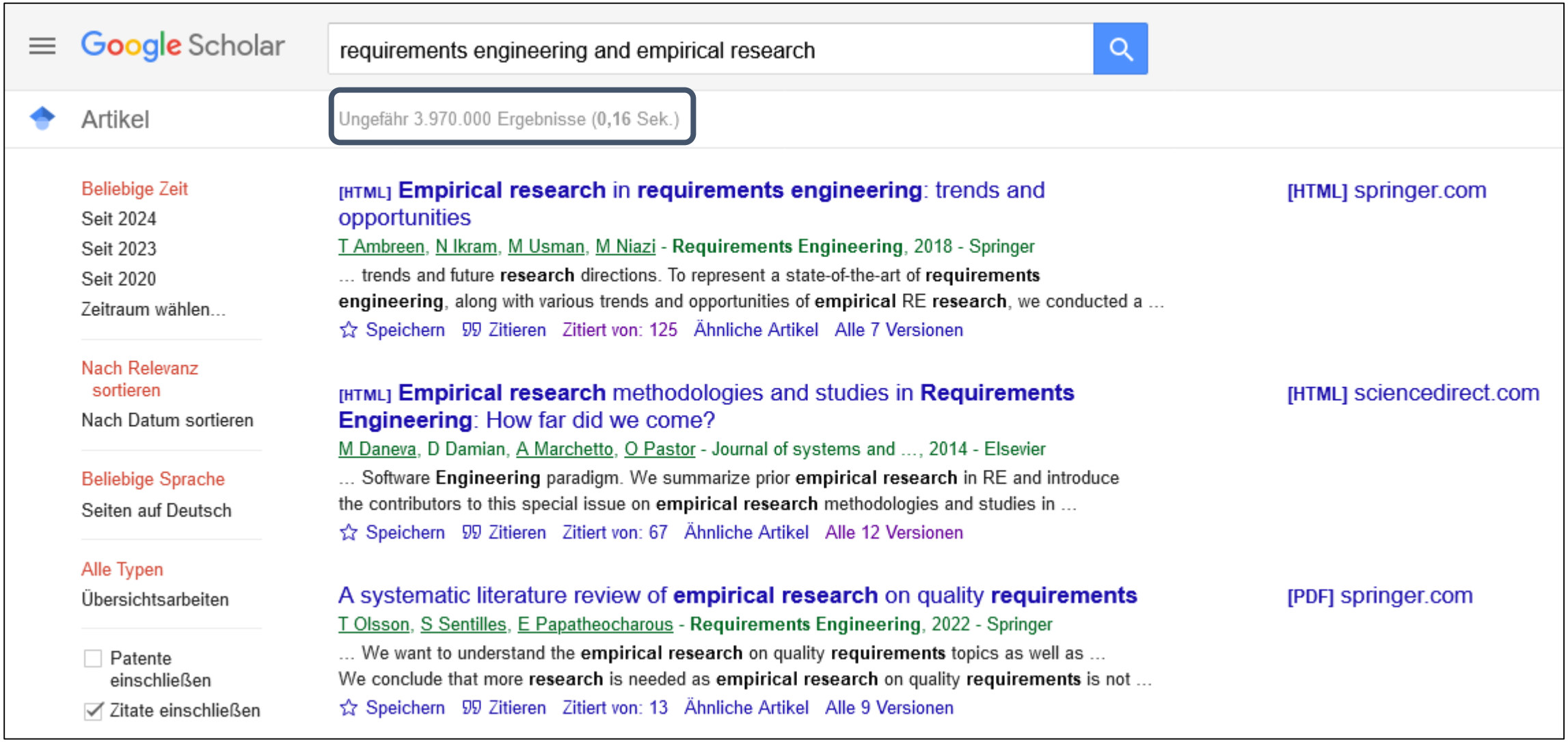
Oren Harari

Digitalization is **more** than just Digitization!

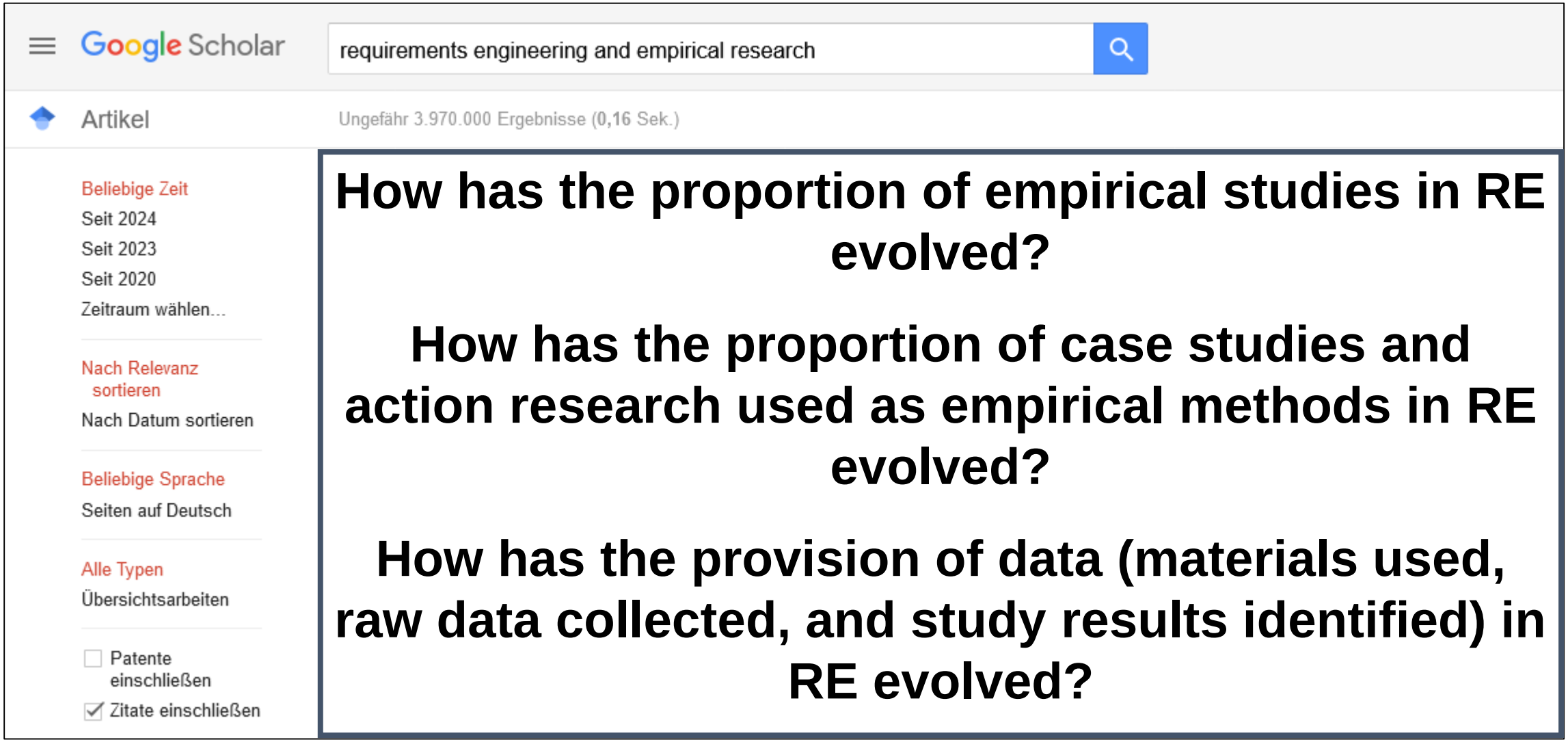
Current and future scientific challenges can not be tackled with an outdated communication system.

Digitalize Knowledge, Not Documents!

Example: Requirements Engineering and Empirical Research



Example: Requirements Engineering and Empirical Research



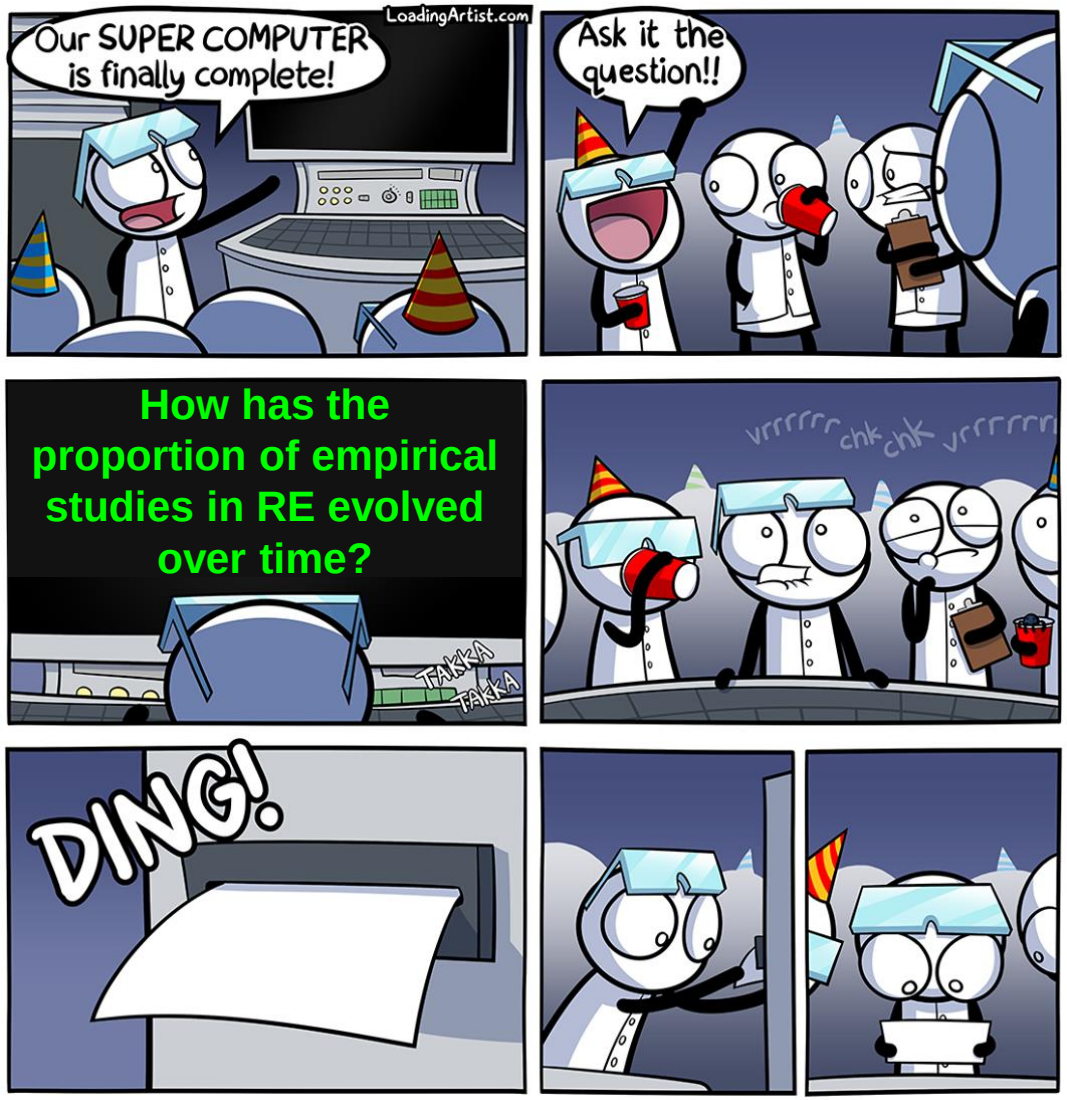
How do We Answer These Questions so far?

(S)**LR**s and SMSs for a **comprehensive**, **up-to-date**, and **long-term available** overview.

Over **7 million publications** per year[4] with an **increasing** (exponential) tendency.

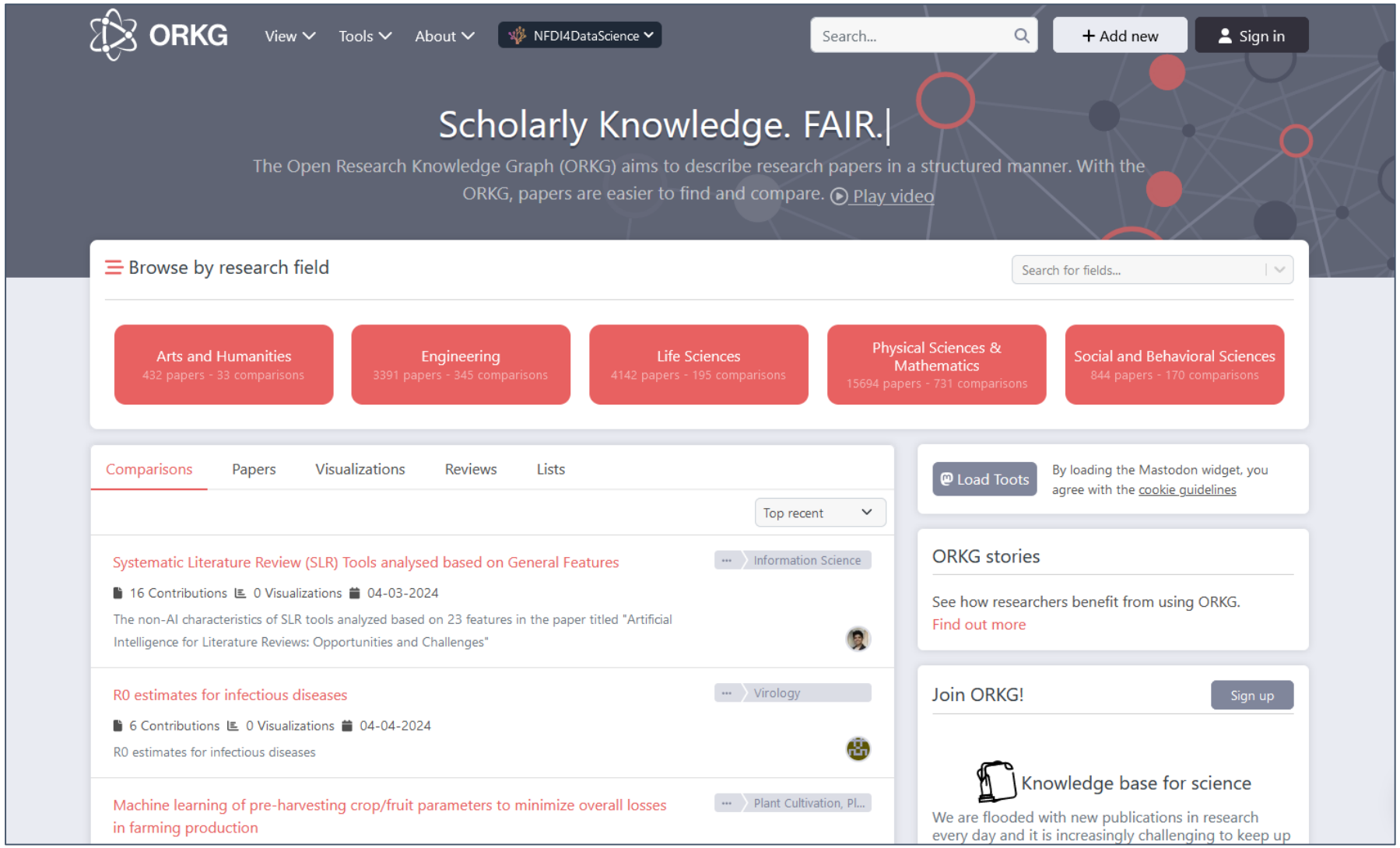
[4] Fire and Guestrin: *Over-Optimization of Academic Publishing Metrics: Observing Goodhart's Law in Action.* GigaScience Vol. 8, No. 6., DOI: 10.1093/gigascience/giz053, 2019.

Wouldn’t it be Great if we Could Ask the Computer?



How can we achieve this goal?

Open Research Knowledge Graph (ORKG)



The ORKG **revolutionizes** how **scientific knowledge** is communicated and processed, making it **actionable** for **machines** and **researchers** to navigate, compare, and review **vast amounts of information** efficiently.

Research Knowledge Graphs

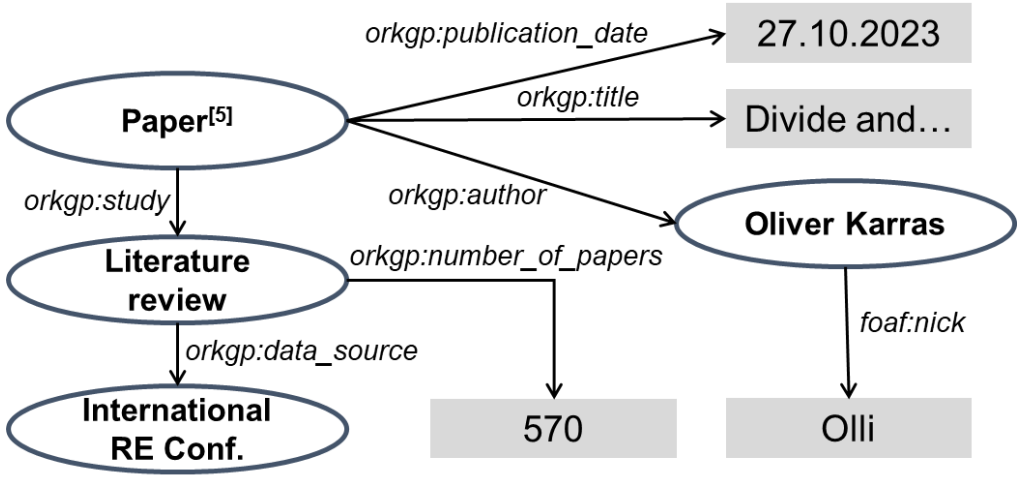
Research Knowledge Graphs (RKGs) are a **technology** for organizing scientific (meta-)data in a

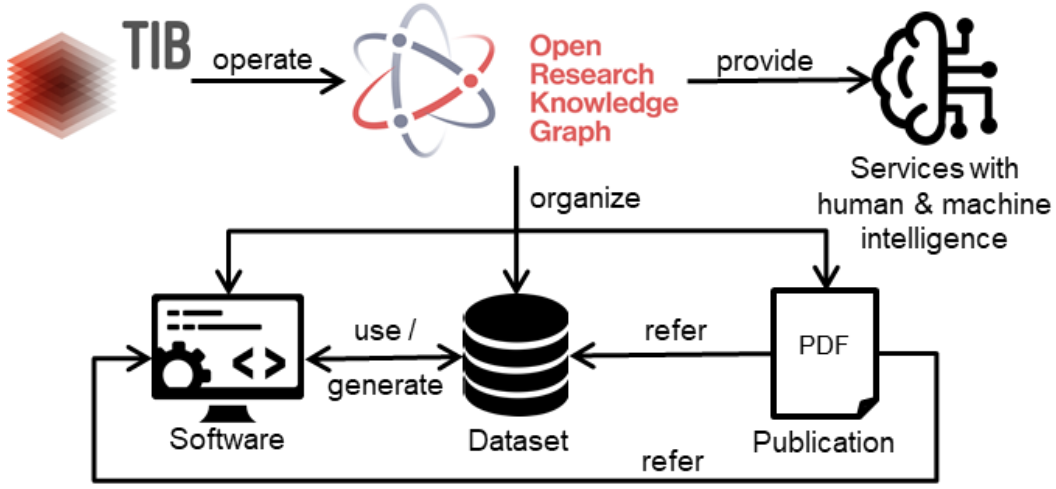
- **Flexible**, **fine-grained**, and **semantic** representation
- That is **understandable** and **processable** by humans and machines

Research Knowledge Graphs

Research Knowledge Graphs (RKGs) are a **technology** for organizing scientific (meta-)data in a

- **Flexible**, **fine-grained**, and **semantic** representation
- That is **understandable** and **processable** by humans and machines

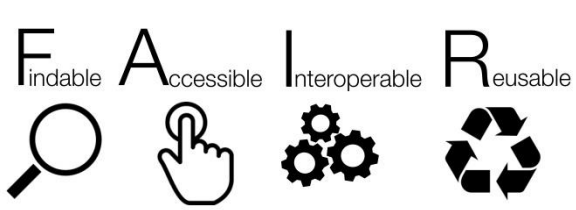




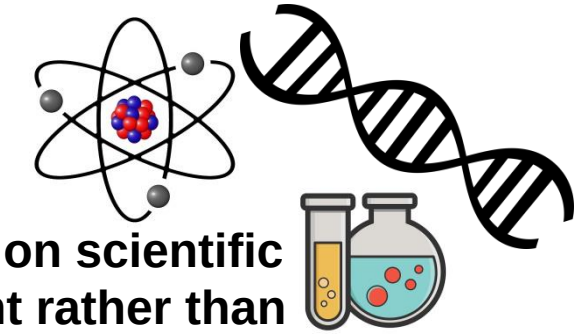
The **ORKG** is a **ready-to-use** and **sustainably** operated **infrastructure** with **services** that uses a **cross-discipline RKG** for the **long-term** and **openly available organization** of scientific (meta-)data according to the **FAIR** data principles.

[5] Karras et al.: *Divide and Conquer the EmpiRE: A Community-Maintainable Knowledge Graph of Empirical Research in Requirements Engineering.* 2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), DOI: 10.1109/ESEM56168.2023.10304795, 2023.

ORKG Objectives



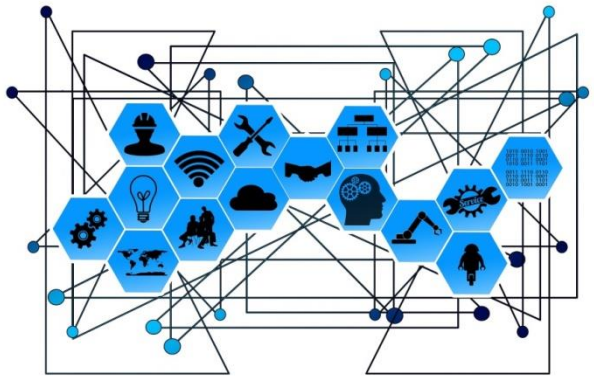
Make research FAIR



Focus on scientific content rather than pure metadata

Foster collaboration

Provide long-term available and up-to-date overviews



Tackle interdisciplinary challenges

Structured Representation of Scientific Knowledge

Mining User Requirements from Application Store Reviews Using Frame Semantics

Nishant Jha and Anas Mahmoud(✉)

The Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA 70803, USA
njha1@lsu.edu, mahmoud@csc.lsu.edu

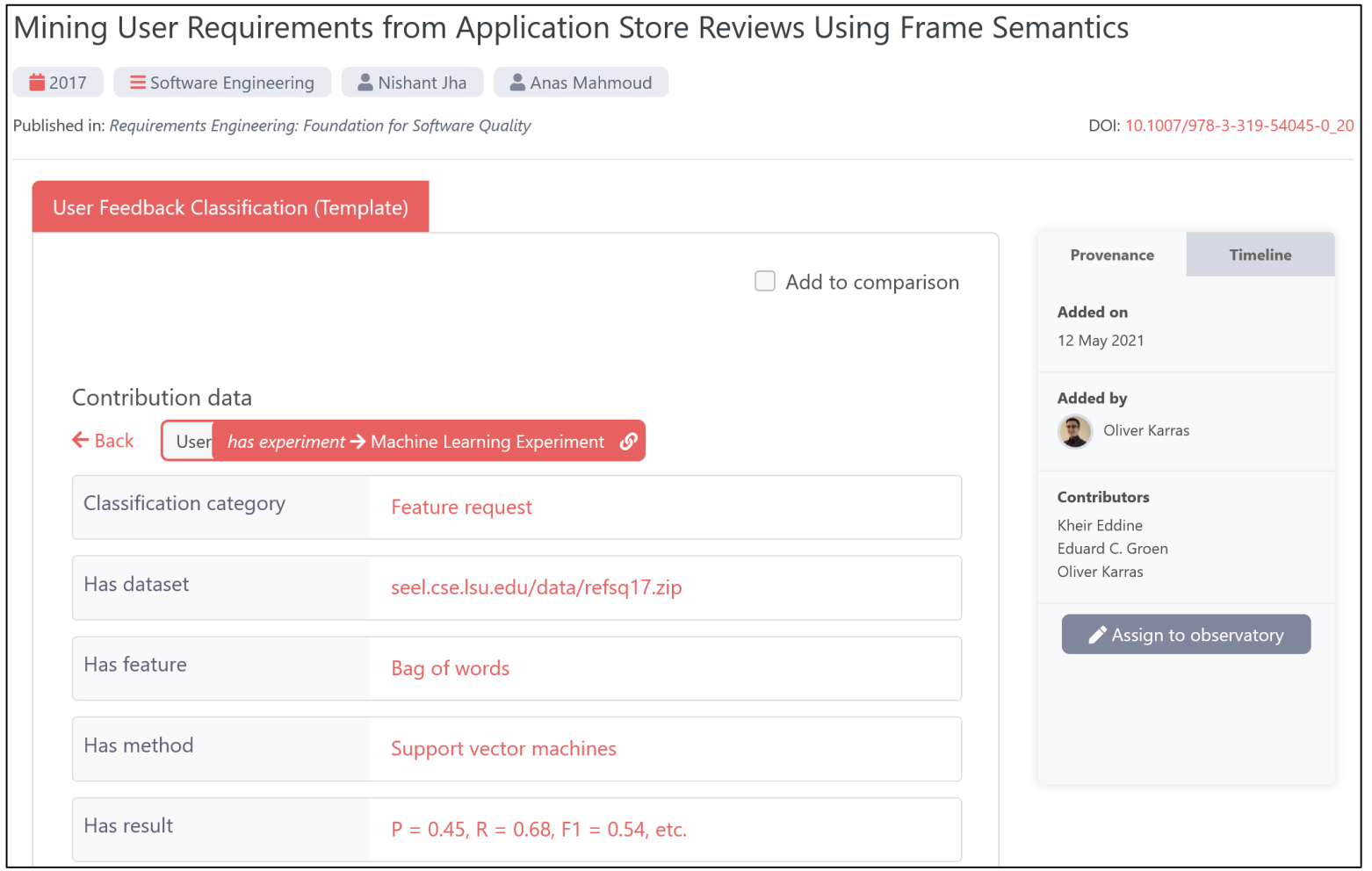
Abstract. *Context and motivation*: Research on mining user reviews in mobile application (app) stores has noticeably advanced in the past few years. The majority of the proposed techniques rely on classifying the textual description of user reviews into different categories of technically informative user requirements and uninformative feedback. *Question/Problem*: Relying on the textual attributes of reviews often produces high dimensional models. This increases the complexity of the classifier and can lead to overfitting problems. *Principal ideas/results*: We propose a novel semantic approach for app review classification. The proposed approach is based on the notion of semantic role labeling, or characterizing the lexical meaning of text in terms of semantic frames. Semantic frames help to generalize from text (individual words) to more abstract scenarios (contexts). This reduces the dimensionality of the data and enhances the predictive capabilities of the classifier. Three datasets of user reviews are used to conduct our experimental analysis. Results show that semantic frames can be used to generate lower dimensional and more accurate models in comparison to text classification methods. *Contribution*: A novel semantic approach for extracting user requirements from app reviews. The proposed approach enables a more efficient classification process and reduces the chance of overfitting.

Keywords: Requirements elicitation · Application stores · Classification

1 Introduction

Mobile application markets, or app stores (e.g., Google Play and Apple App Store), represent a unique model of service-oriented business. Such platforms have created an unprecedented opportunity for app developers to directly monitor the opinions of a large population of end-users of their software [25]. Through app stores feedback services, app users can directly share their experience in the form of textual reviews and meta-data (e.g., star ratings). Analyzing large datasets of app store reviews has revealed that they contain substantial amounts of up-to-date technical information. Such information can be leveraged by app developers to help them maintain and sustain their apps in a highly-competitive

P. Grünbacher and A. Perini (Eds.): REFSQ 2017, LNCS 10153, pp. 273–287, 2017.
DOI: 10.1007/978-3-319-54045-0_20



Behind the Scenes

Mining User Requirements from Application Store Reviews Using Frame Semantics

Nishant Jha and Anas Mahmoud(✉)

The Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA 70803, USA
njha1@lsu.edu, mahmoud@csc.lsu.edu

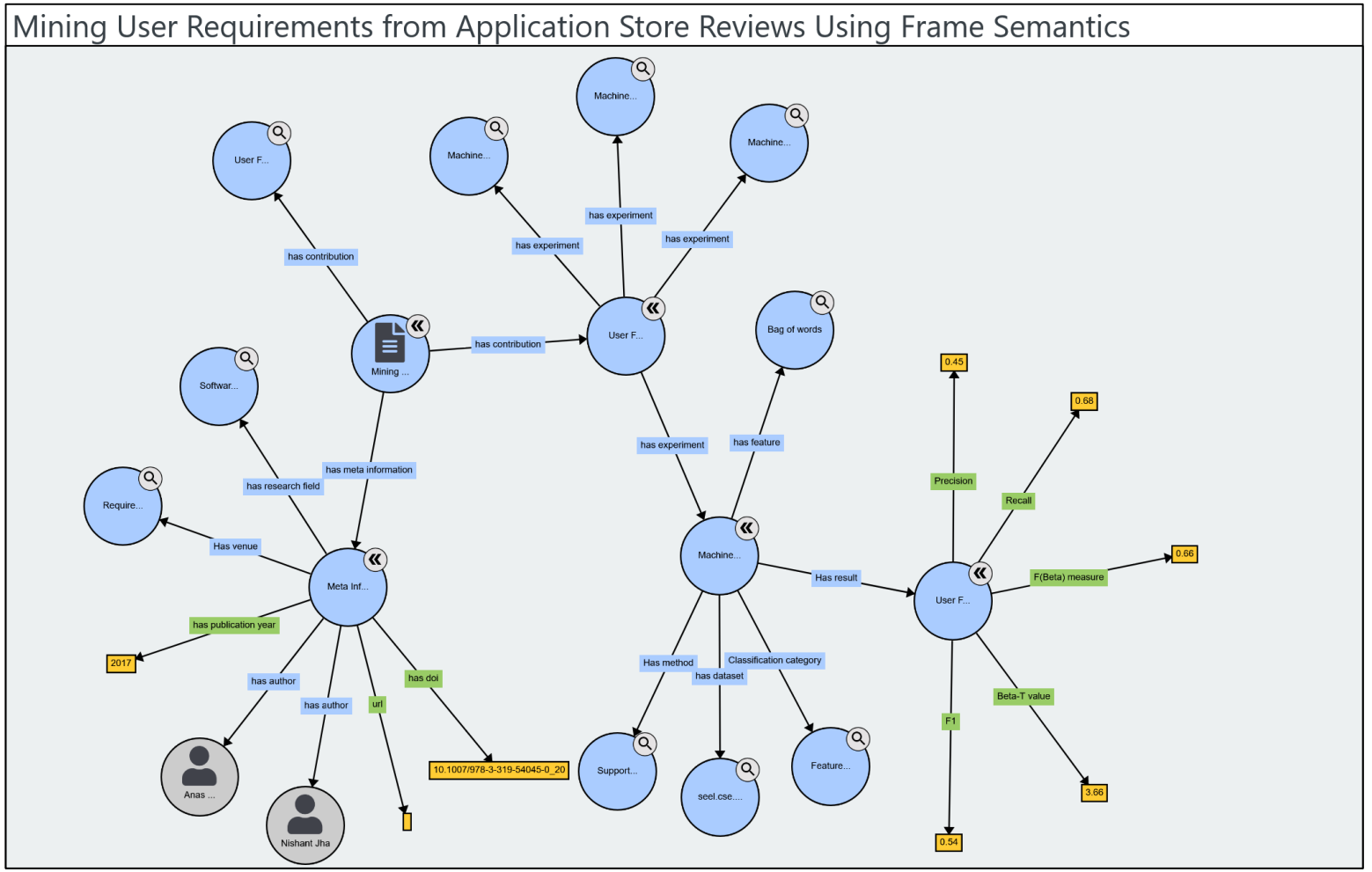
Abstract. *Context and motivation*: Research on mining user reviews in mobile application (app) stores has noticeably advanced in the past few years. The majority of the proposed techniques rely on classifying the textual description of user reviews into different categories of technically informative user requirements and uninformative feedback. *Question/Problem*: Relying on the textual attributes of reviews often produces high dimensional models. This increases the complexity of the classifier and can lead to overfitting problems. *Principal ideas/results*: We propose a novel semantic approach for app review classification. The proposed approach is based on the notion of semantic role labeling, or characterizing the lexical meaning of text in terms of semantic frames. Semantic frames help to generalize from text (individual words) to more abstract scenarios (contexts). This reduces the dimensionality of the data and enhances the predictive capabilities of the classifier. Three datasets of user reviews are used to conduct our experimental analysis. Results show that semantic frames can be used to generate lower dimensional and more accurate models in comparison to text classification methods. *Contribution*: A novel semantic approach for extracting user requirements from app reviews. The proposed approach enables a more efficient classification process and reduces the chance of overfitting.

Keywords: Requirements elicitation · Application stores · Classification

1 Introduction

Mobile application markets, or app stores (e.g., Google Play and Apple App Store), represent a unique model of service-oriented business. Such platforms have created an unprecedented opportunity for app developers to directly monitor the opinions of a large population of end-users of their software [25]. Through app stores feedback services, app users can directly share their experience in the form of textual reviews and meta-data (e.g., star ratings). Analyzing large datasets of app store reviews has revealed that they contain substantial amounts of up-to-date technical information. Such information can be leveraged by app developers to help them maintain and sustain their apps in a highly-competitive

P. Grünbacher and A. Perini (Eds.): REFSQ 2017, LNCS 10153, pp. 273–287, 2017.
DOI: 10.1007/978-3-319-54045-0_20



Using FAIR Scientific Knowledge

Mining User Requirements from Application Store Reviews Using Frame Semantics

Nishant Jha and Anas Mahmoud(✉)

The Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA 70803, USA
njha1@lsu.edu, mahmoud@csc.lsu.edu

Abstract. *Context and motivation*: Research on mining user reviews in mobile application (app) stores has noticeably advanced in the past few years. The majority of the proposed techniques rely on classifying the textual description of user reviews into different categories of technically informative user requirements and uninformative feedback. *Question/Problem*: Relying on the textual attributes of reviews often produces high dimensional models. This increases the complexity of the classifier and can lead to overfitting problems. *Principal ideas/results*: We propose a novel semantic approach for app review classification. The proposed approach is based on the notion of semantic role labeling, or characterizing the lexical meaning of text in terms of semantic frames. Semantic frames help to generalize from text (individual words) to more abstract scenarios (contexts). This reduces the dimensionality of the data and enhances the predictive capabilities of the classifier. Three datasets of user reviews are used to conduct our experimental analysis. Results show that semantic frames can be used to generate lower dimensional and more accurate models in comparison to text classification methods. *Contribution*: A novel semantic approach for extracting user requirements from app reviews. The proposed approach enables a more efficient classification process and reduces the chance of overfitting.

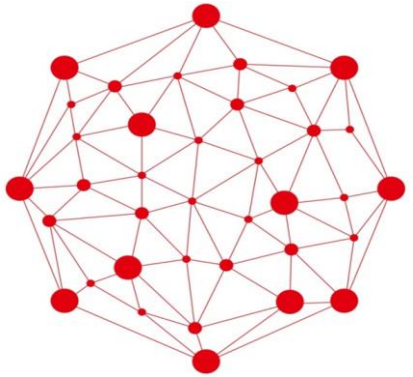
Keywords: Requirements elicitation · Application stores · Classification

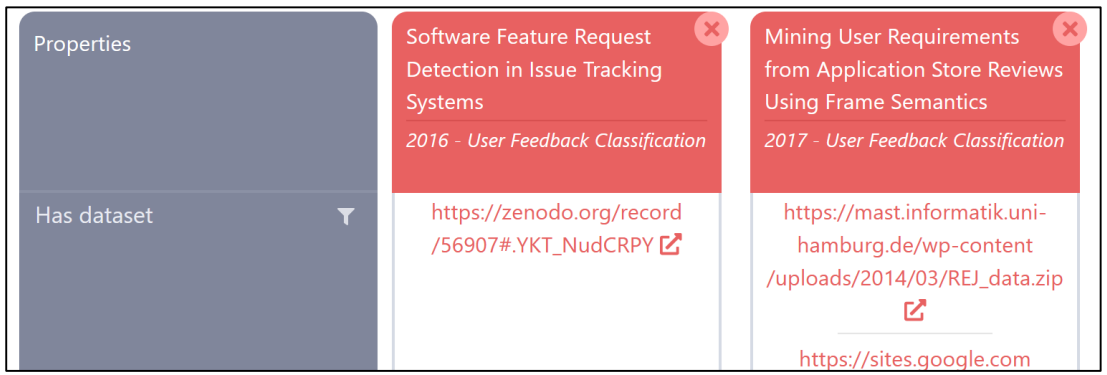
1 Introduction

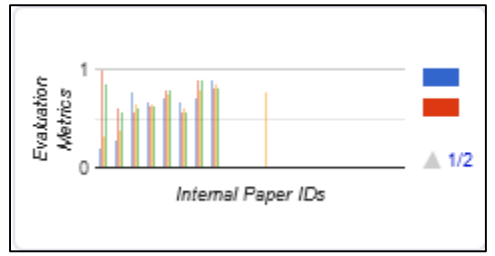
Mobile application markets, or app stores (e.g., Google Play and Apple App Store), represent a unique model of service-oriented business. Such platforms have created an unprecedented opportunity for app developers to directly monitor the opinions of a large population of end-users of their software [25]. Through app stores feedback services, app users can directly share their experience in the form of textual reviews and meta-data (e.g., star ratings). Analyzing large datasets of app store reviews has revealed that they contain substantial amounts of up-to-date technical information. Such information can be leveraged by app developers to help them maintain and sustain their apps in a highly-competitive

P. Grünbacher and A. Perini (Eds.): REFSQ 2017, LNCS 10153, pp. 273–287, 2017.
DOI: 10.1007/978-3-319-54045-0_20

How has the provision of data (materials used, raw data collected, and study results identified) in RE evolved?

Natural language question →  → **Answer**





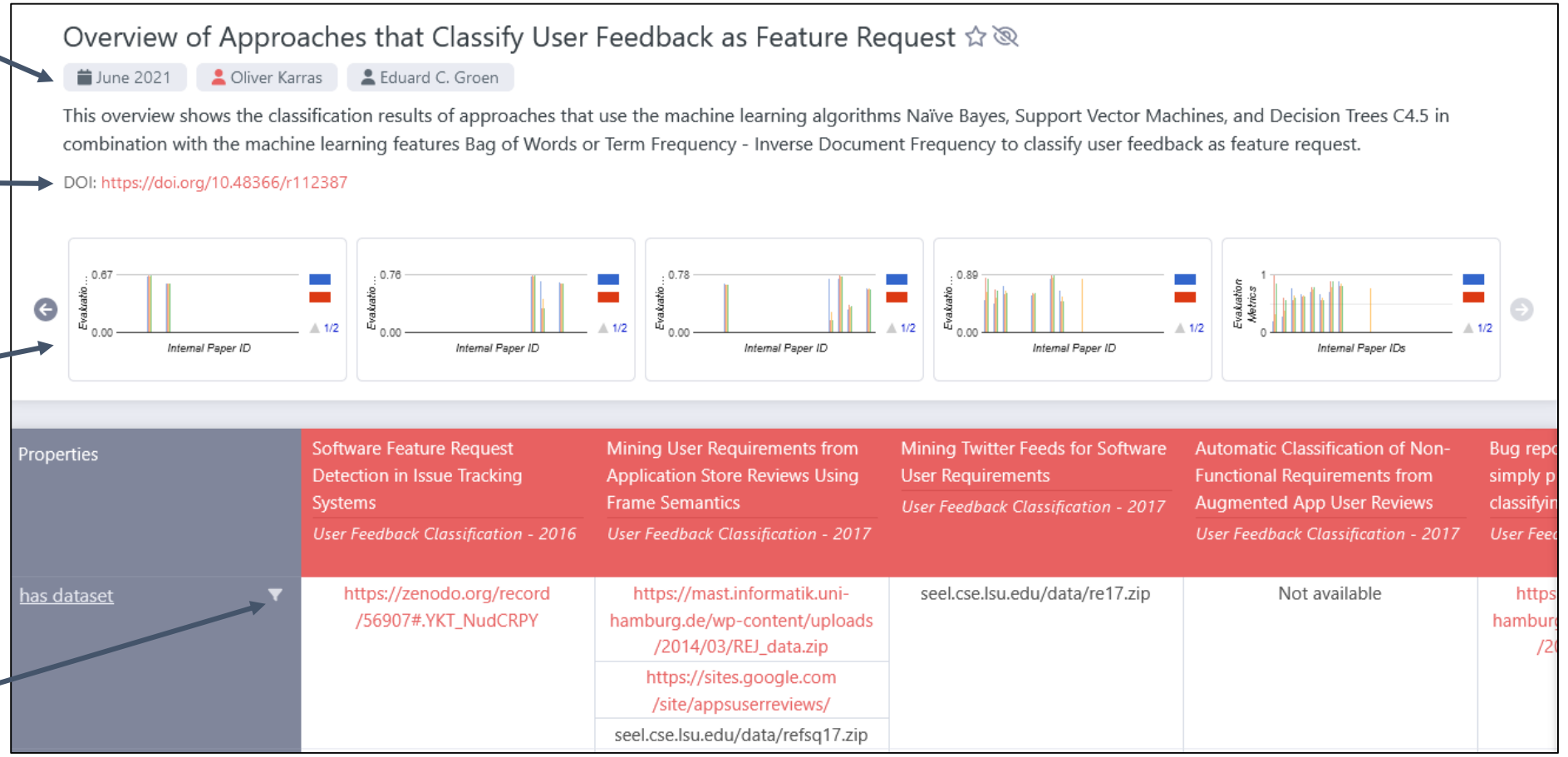
ORKG Comparisons

Acknowledgement of creators

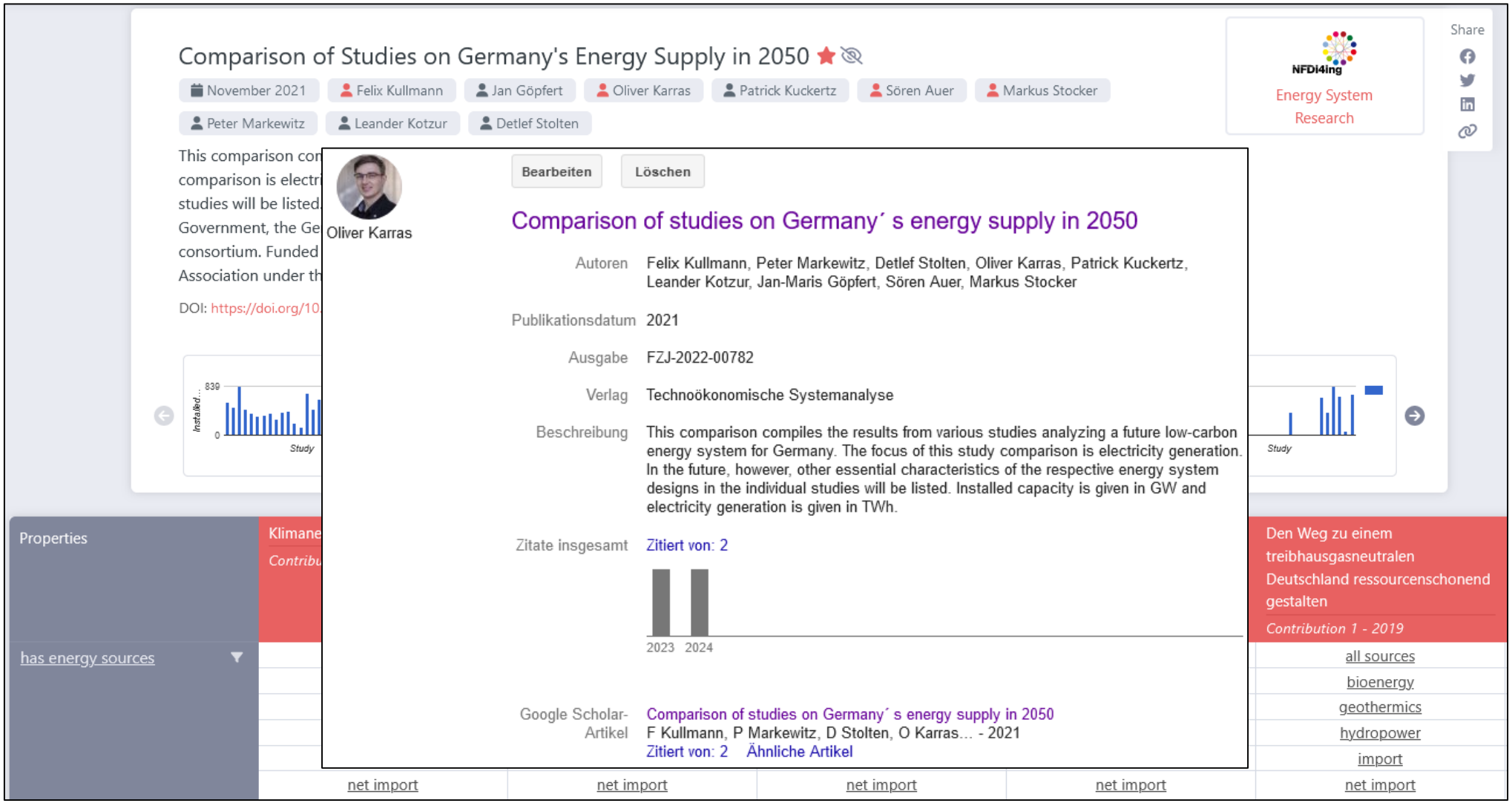
DOI

Visualizations

Interactive filtering



ORKG Comparisons are Citable



So far so Good, but…

…what can we do with machine-actionable scientific knowledge?

Anything we want!

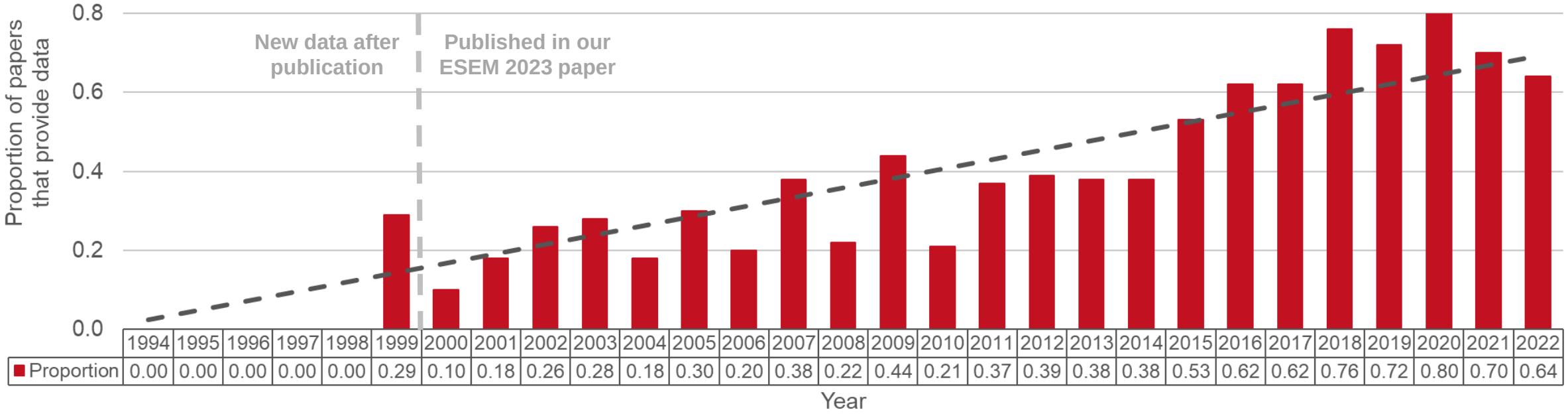
1. Papers, comparisons, and visualizations in the ORKG are **openly available for (re-)use and extension to anyone.**
2. The ORKG provides several **interfaces** for processing the data, e.g., to develop novel search, retrieval, mining, and assistance applications.

{ REST : API }

(Re-)use and Extension of Scientific Knowledge

How has the provision of data (materials used, raw data collected, and study results identified) in RE evolved?[5, 6, 7]

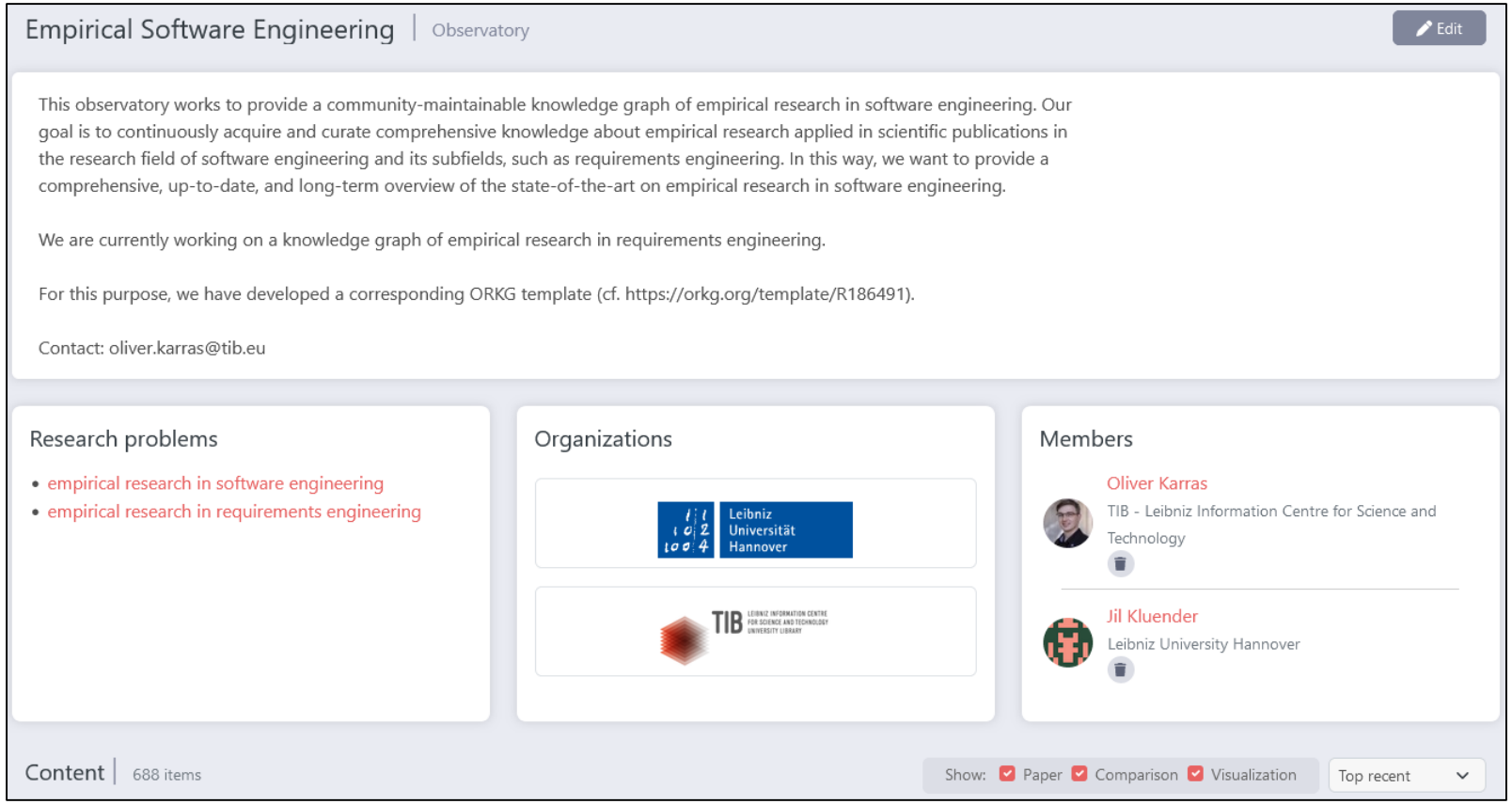
(Based on 680 papers in the ORKG)



	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
Proportion	0.00	0.00	0.00	0.00	0.00	0.29	0.10	0.18	0.26	0.28	0.18	0.30	0.20	0.38	0.22	0.44	0.21	0.37	0.39	0.38	0.38	0.53	0.62	0.62	0.76	0.72	0.80	0.70	0.64

[5] Karras et al.: *Divide and Conquer the EmpiRE: A Community-Maintainable Knowledge Graph of Empirical Research in Requirements Engineering.* 2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), DOI: 10.1109/ESEM56168.2023.10304795, 2023.
[6] Project on GitHub: https://github.com/okarras/EmpiRE-Analysis
[7] Interactive Jupyter notebook: https://mybinder.org/v2/gh/okarras/EmpiRE-Analysis/HEAD?labpath=%2Fempire-analysis.ipynb

ORKG Observatory



- **Open** groups maintain topics in the ORKG
- **Central** access point for the community to all curated contents

Who organizes this scientific knowledge in the ORKG?

Who Organizes this Scientific Knowledge in the ORKG?

The heart of the ORKG is its **CROWD**:

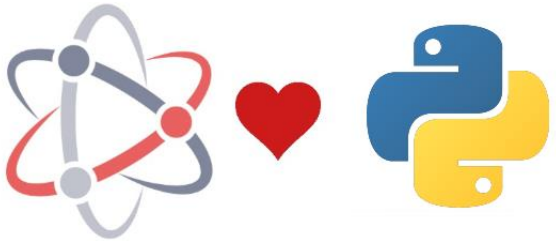
Researchers from any discipline!

How can We Contribute to the ORKG?

1. While writing a paper:
   - SciKGTeX: LaTeX package for FAIR annotations in a paper (embedded in the PDF) that can be imported into ORKG
   - https://github.com/Christof93/SciKGTeX

SciKGTeX

2. While developing an analysis:
   - Python package (& R package in development) for FAIR annotations in analysis scripts that can be imported into the ORKG
   - https://pypi.org/project/orkg/



3. At any time:
   - Manually using the ORKG Frontend to describe papers
   - (Semi-)automatically using the ORKG REST API

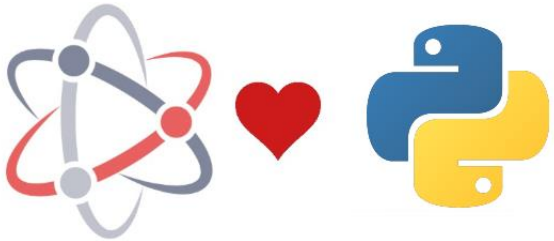
Open Research Knowledge Graph { REST : API }

What We Will Learn to Contribute to the ORKG

1. While writing a paper:
   - **SciKGTeX**: LaTeX package for FAIR annotations in a paper (embedded in the PDF) that can be imported into ORKG
   - https://github.com/Christof93/SciKGTeX

SciKGTeX

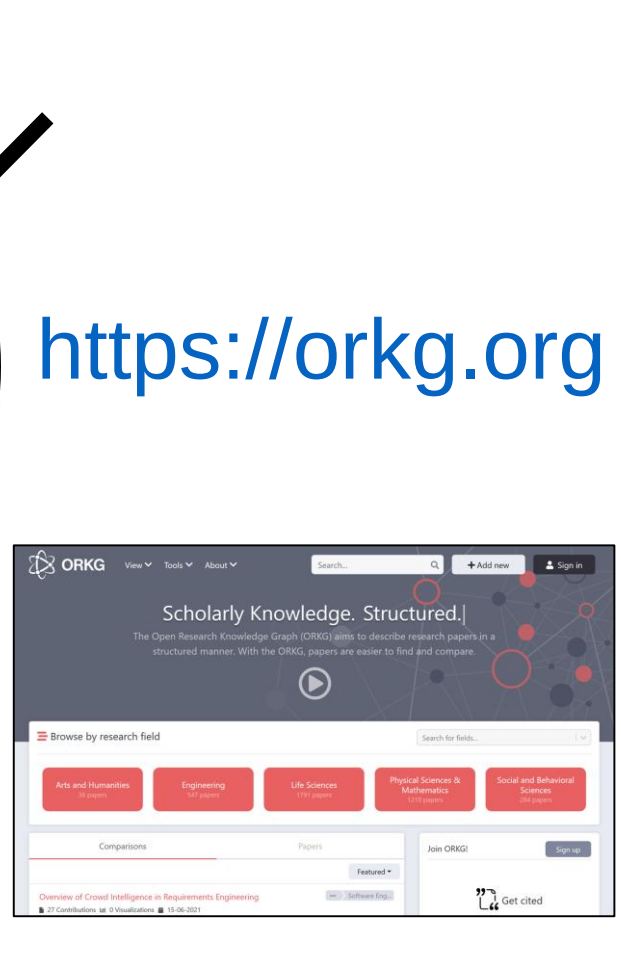
2. While developing an analysis:
   - Python package (& R package in development) for FAIR annotations in analysis scripts that can be imported into the ORKG
   - https://pypi.org/project/orkg/



3. At any time:
   - **Manually** using the ORKG Frontend to describe papers
   - (Semi-)automatically using the ORKG REST API

Open Research Knowledge Graph { REST : API }

Conclusion



https://orkg.org

Let's bring scholarly communication and open science in Requirements Engineering to the 21st century!

Further Reading

Open Access ORKG Book

- Celebrating the 5th anniversary of ORKG
- A practical guide for new and advanced users
  - ORKG's terms and concepts
  - ORKG's approach
  - ORKG's technology
  - ORKG's success stories

https://cuvillier.de/de/shop/publications/9037-open-research-knowledge-graph

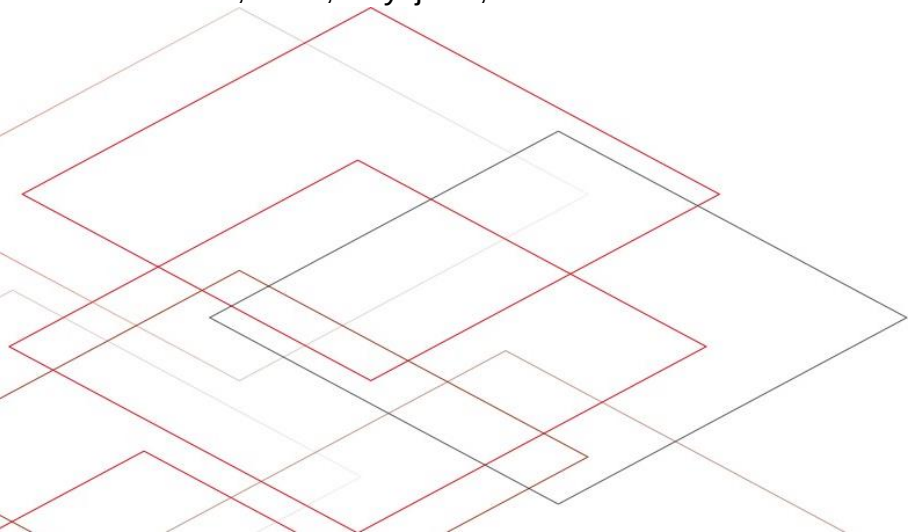
SciKGTeX:
A LaTeX Package for FAIR-Annotated Publications

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

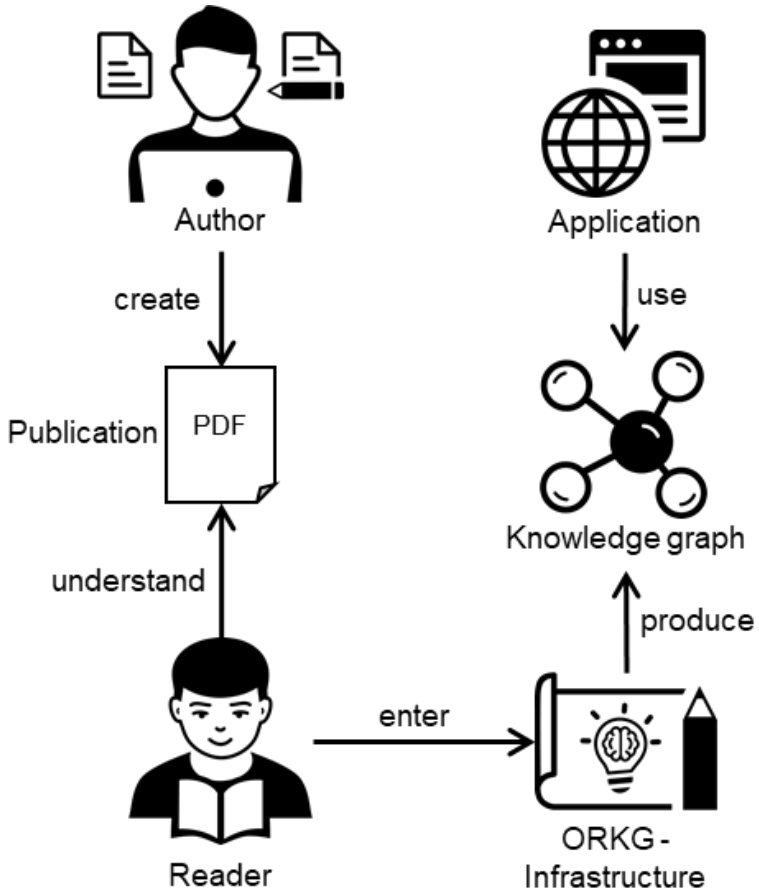
32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

June 24th, 2024, Reykjavik, Iceland



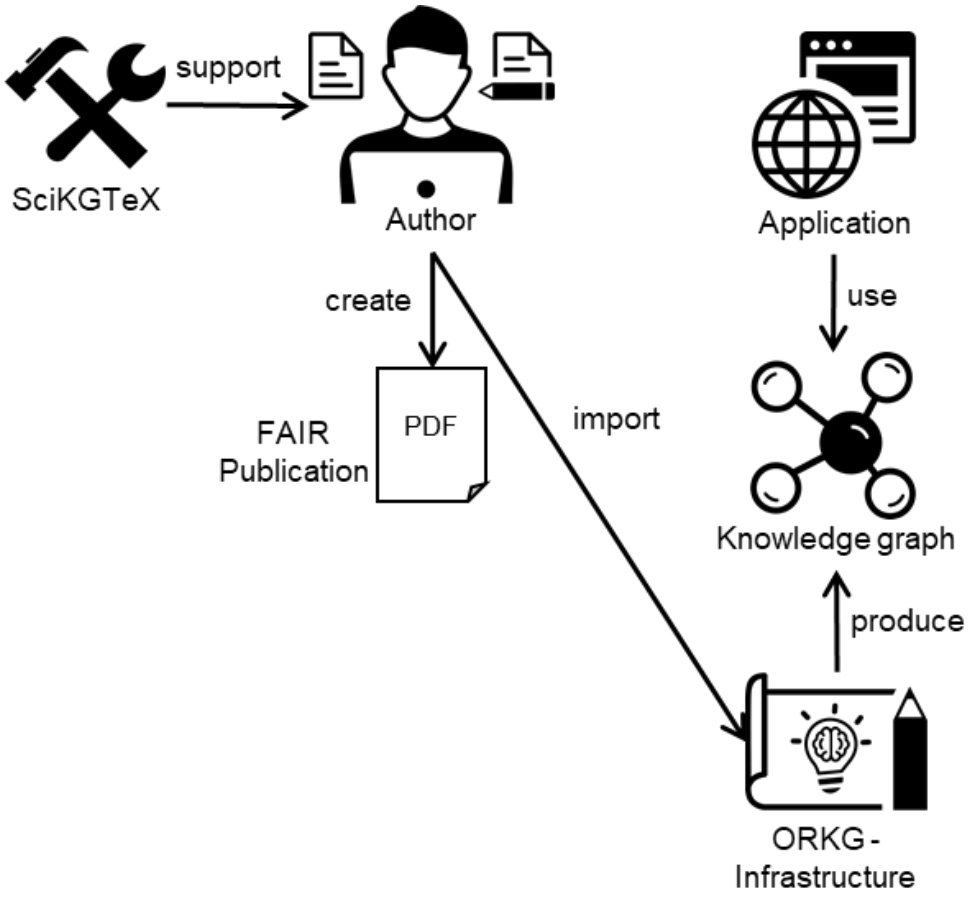
Problem: Making Publications FAIR is a Downstream Task

Problem: ORKG focuses on **published** articles, so making them **FAIR** is a **downstream task** and often **not done by the author**.



Solution: “FAIR-by-Design” Artifacts and SciKGTeX[8]

Solution: “FAIR-by-Design” Artifacts. **SciKGTeX** supports authors in making their publication **FAIR** at the time of **creation**.



- **Authors describe** their publication with FAIR information **only once** and **in parallel** at the time of **creation**
- **SciKGTeX embeds** FAIR information **into** the **PDF metadata** as a knowledge graph
  - **Persistent** over PDF lifetime
  - **Available** for anyone
  - **Reusable**, e.g., import into ORKG

[8] Bless et al.: *SciKGTeX – A LaTeX Package to Semantically Annotate Contributions in Scientific Publications.* 2023 ACM/IEEE Joint Conference on Digital Libraries (JCDL), DOI: 10.1109/JCDL57899.2023.00030, 2023.

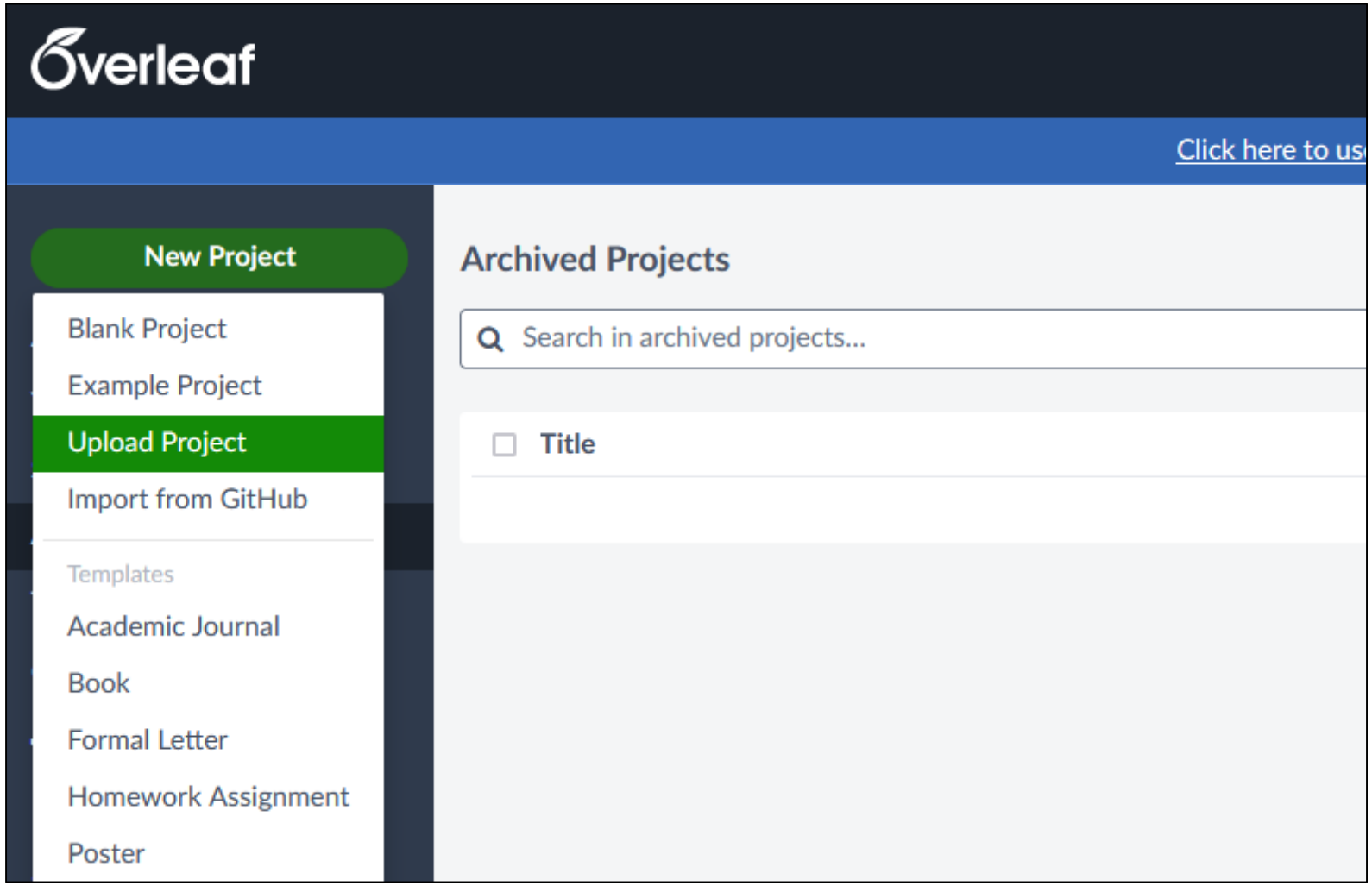
SciKGTeX – Scientific Knowledge Graph TeX

- Predefined commands for annotation
  - 3 commands for metadata
    - Title
    - Author
    - Research field
  - 5 commands for content
    - Research problem
    - Objective
    - Method
    - Result
    - Conclusion
- Support for own custom annotations
  - REFSQ'24 and REFSQ'25 ask for
    - Code repository
    - Dataset

```
\usepackage{scikgtex}
\begin{document}
The role of \researchproblem{antibiotic therapy} is controversial.
The purpose of this study was to \objective{determine the
effectiveness of high-dose amoxicillin/potassium clavulanate in the
treatment of children}.
This was a \method{randomized, double-blind, placebo-controlled
study}.
\result{Children receiving the antibiotic were more likely to be
cured (50% vs 14%) than children receiving the placebo}.
\conclusion{Amoxicillin/potassium clavulanate results in
significantly more cures and fewer failures than placebo}.
\end{document}
```

Complete documentation and latest version:
https://github.com/Christof93/SciKGTeX

Set up a LaTeX Project (using Overleaf)



1. Download Example.zip

https://bit.ly/49RjRbO

2. Open Overleaf

https://www.overleaf.com/

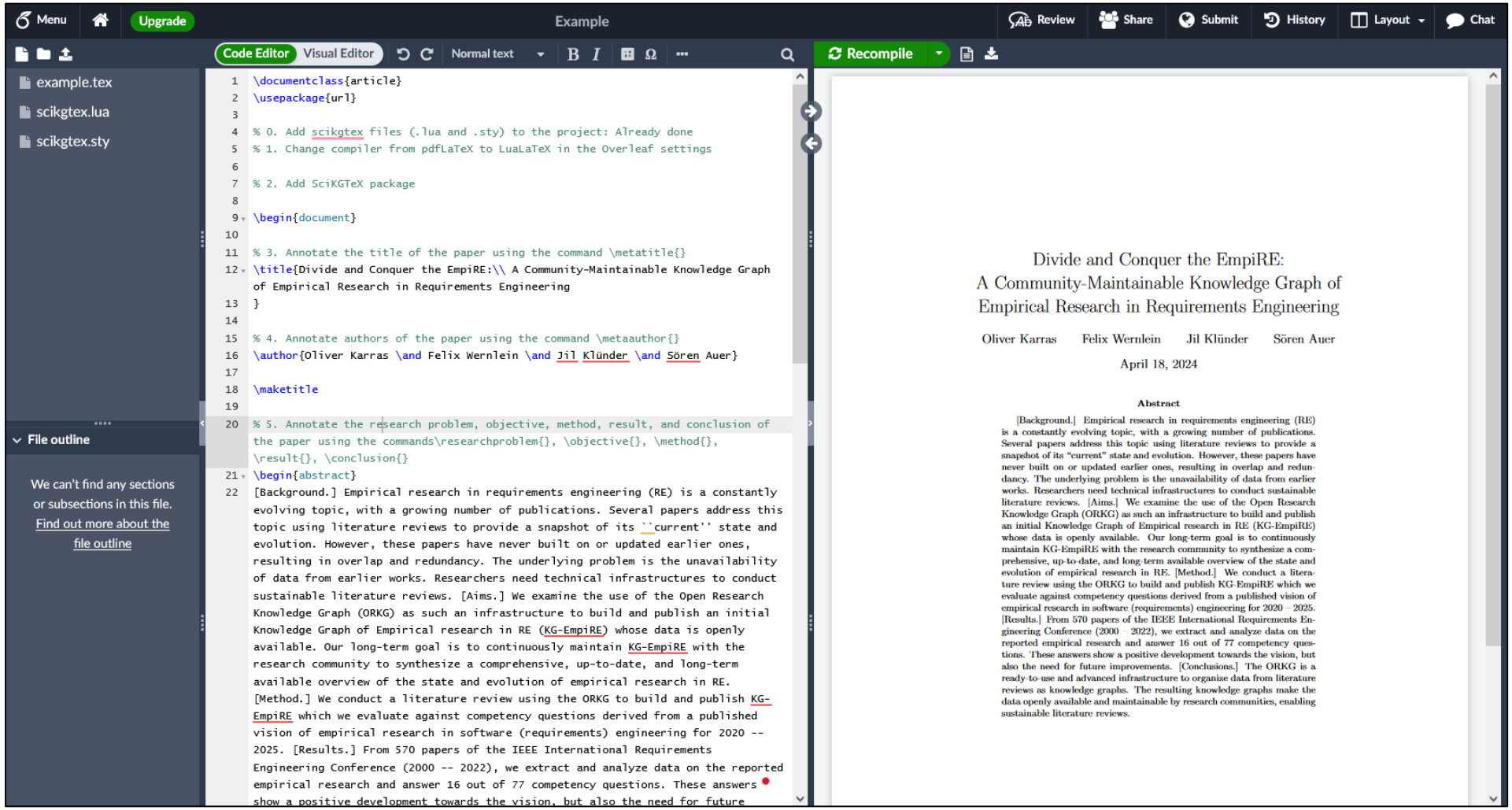
3. Upload Example.zip

New Project → Upload Project

4. Select or drag zip file

Remark:
We only use an abstract for the annotations as a simplified example. Annotations can be used **anywhere** (in one or more LaTeX files).

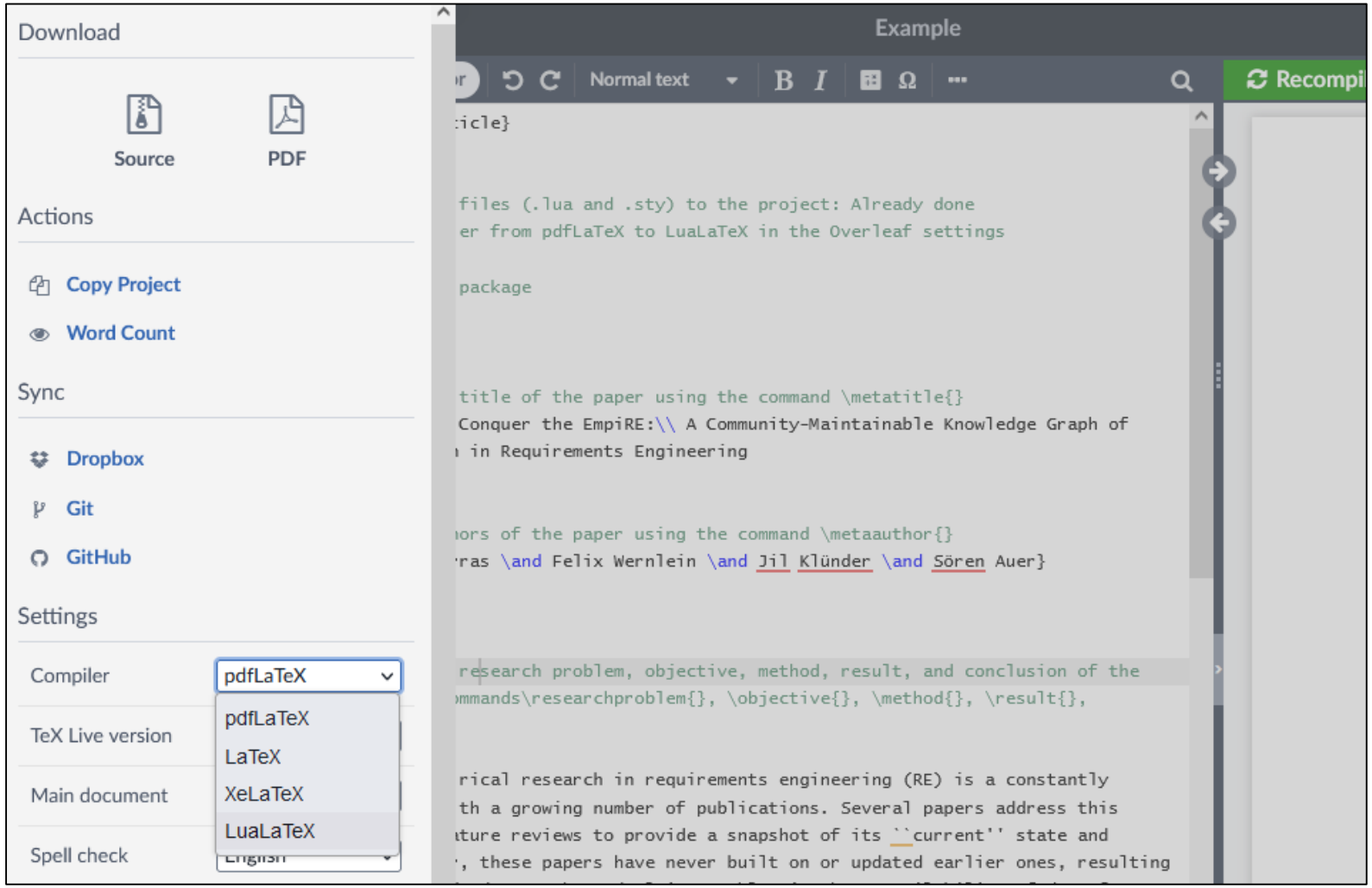
0. Add SciKGTeX Files to the Project



The following two files must be present:

- scikgtex.lua
- scikgtex.sty

1. Change Complier from pdfLaTeX to LuaLaTeX



It is necessary to compile your LaTeX source with **LuaLaTeX** for the SciKGTeX package to work.

2. Add the SciKGTeX Package to the LaTeX File

```
1  \documentclass{article}
2  \usepackage{url}
3
4  % 0. Add scikgtex files (.lua and .sty) to the project: Already done
5  % 1. Change compiler from pdfLaTeX to LuaLaTeX in the Overleaf settings
6
7  % 2. Add SciKGTeX package
8
9
10 \begin{document}
11
```

Add the command to the preamble of the LaTeX file:

\usepackage{scikgtex}

2. Add the SciKGTeX Package to the LaTeX File: Result

```
1  \documentclass{article}
2  \usepackage{url}
3
4  % 0. Add scikgtex files (.lua and .sty) to the project: Already done
5  % 1. Change compiler from pdfLaTeX to LuaLaTeX in the Overleaf settings
6
7  % 2. Add SciKGTeX package
8  \usepackage{scikgtex}
9
10 \begin{document}
11
```

Add the command to the preamble of the LaTeX file:

\usepackage{scikgtex}

3. & 4. Annotate Metadata of the Paper

```
10  \begin{document}
11
12  % 3. Annotate the title of the paper using the command \metatitle{}
13  \title{Divide and Conquer the EmpiRE:\\ A Community-Maintainable Knowledge Graph of
    Empirical Research in Requirements Engineering
14  }
15
16  % 4. Annotate authors of the paper using the command \metaauthor{}
17  \author{Oliver Karras \and Felix Wernlein \and Jil Klünder \and Sören Auer}
18
19  \maketitle
20
```

1. Annotate the title

\metatitle{}

2. Annotate the authors

\metaauthor{}

Remark:
The command \metaauthor{} must be used individually **for each author**. With four authors, you need the command four times.

3. & 4. Annotate Metadata of the Paper: Result

```
10  \begin{document}
11
12  % 3. Annotate the title of the paper using the command \metatitle{}
13  \title{\metatitle{Divide and Conquer the EmpiRE:\\ A Community-Maintainable Knowledge
    Graph of Empirical Research in Requirements Engineering}
14  }
15
16  % 4. Annotate authors of the paper using the command \metaauthor{}
17  \author{\metaauthor{Oliver Karras} \and \metaauthor{Felix Wernlein} \and
    \metaauthor{Jil Klünder} \and \metaauthor{Sören Auer}}
18
19  \maketitle
20
```

1. Annotate the title

\metatitle{}

2. Annotate the authors

\metaauthor{}

Remark:
The command \metaauthor{} must be used individually **for each author**. With four authors, you need the command four times.

5. Annotate Content of the Paper

```
21  % 5. Annotate the research problem, objective, method, result, and conclusion of the
    paper using the commands\researchproblem{}, \objective{}, \method{}, \result{},
    \conclusion{}
22  \begin{abstract}
23  [Background.] Empirical research in requirements engineering (RE) is a constantly
    evolving topic, with a growing number of publications. Several papers address this
    topic using literature reviews to provide a snapshot of its ``current'' state and
    evolution. However, these papers have never built on or updated earlier ones, resulting
    in overlap and redundancy. The underlying problem is the unavailability of data from
    earlier works. Researchers need technical infrastructures to conduct sustainable
    literature reviews. [Aims.] We examine the use of the Open Research Knowledge Graph
    (ORKG) as such an infrastructure to build and publish an initial Knowledge Graph of
    Empirical research in RE (KG-EmpiRE) whose data is openly available. Our long-term goal
    is to continuously maintain KG-EmpiRE with the research community to synthesize a
    comprehensive, up-to-date, and long-term available overview of the state and evolution
    of empirical research in RE. [Method.] We conduct a literature review using the ORKG to
    build and publish KG-EmpiRE which we evaluate against competency questions derived from
    a published vision of empirical research in software (requirements) engineering for
    2020 -- 2025. [Results.] From 570 papers of the IEEE International Requirements
    Engineering Conference (2000 -- 2022), we extract and analyze data on the reported
    empirical research and answer 16 out of 77 competency questions. These answers show a
    positive development towards the vision, but also the need for future improvements.
    [Conclusions.] The ORKG is a ready-to-use and advanced infrastructure to organize data
    from literature reviews as knowledge graphs. The resulting knowledge graphs make the
    data openly available and maintainable by research communities, enabling sustainable
    literature reviews.
24  \end{abstract}
```

1. Annotate the research problem

\researchproblem{}

2. Annotate the objective

\objective{}

3. Annotate the method

\method{}

4. Annotate the result

\result{}

5. Annotate the conclusion

\conclusion{}

5. Annotate Content of the Paper

```
21  % 5. Annotate the research problem, objective, method, result, and conclusion of the
```

Remark:

- All commands can be used **multiple** times.
- The annotated text elements should be as **short** as **possible** and as **long** as **necessary**.
- Definitions:
  - **Research problem**: Issue or gap in existing knowledge addressed by the paper.
  - **Objective**: Goal that the paper aims to achieve.
  - **Method**: Systematic approach, technique, or action plan used in the paper to achieve a goal and result.
  - **Result**: Outcome from a systematic approach, technique, or action plan used in the paper.
  - **Conclusion**: Findings from the analysis of the research results in the paper.

```
24  \end{abstract}
```

1. Annotate the research problem

\researchproblem{}

2. Annotate the objective

\objective{}

3. Annotate the method

\method{}

4. Annotate the result

\result{}

5. Annotate the conclusion

\conclusion{}

5. Annotate Content of the Paper: Result

```
% 5. Annotate the research problem, objective, method, result, and conclusion of the paper using the commands\researchproblem{}, \objective{}, \method{}, \result{}, \conclusion{}
\begin{abstract}
[Background.] Empirical research in requirements engineering (RE) is a constantly evolving topic, with a growing number of publications. Several papers address this topic using literature reviews to provide a snapshot of its ``current'' state and evolution. However, these papers have never built on or updated earlier ones, resulting in overlap and redundancy. The underlying problem is the \researchproblem{unavailability of data from earlier works}. Researchers need technical infrastructures to conduct sustainable literature reviews. [Aims.] We examine the \objective{use of the Open Research Knowledge Graph (ORKG) as such an infrastructure to build and publish an initial Knowledge Graph of Empirical research in RE (KG-EmpiRE) whose data is openly available}. Our long-term goal is to continuously maintain KG-EmpiRE with the research community to synthesize a comprehensive, up-to-date, and long-term available overview of the state and evolution of empirical research in RE. [Method.] We conduct a \method{literature review using the ORKG} to build and publish KG-EmpiRE which we \method{evaluate against competency questions} derived from a published vision of empirical research in software (requirements) engineering for 2020 -- 2025. [Results.] \result{From 570 papers of the IEEE International Requirements Engineering Conference (2000 -- 2022), we extract and analyze data on the reported empirical research} and \result{answer 16 out of 77 competency questions}. These answers show a positive development towards the vision, but also the need for future improvements. [Conclusions.] \conclusion{The ORKG is a ready-to-use and advanced infrastructure to organize data from literature reviews as knowledge graphs}. The resulting knowledge graphs make the data openly available and maintainable by research communities, enabling sustainable literature reviews.
\end{abstract}
```

1. Annotate the research problem

\researchproblem{}

2. Annotate the objective

\objective{}

3. Annotate the method

\method{}

4. Annotate the result

\result{}

5. Annotate the conclusion

\conclusion{}

6. Annotate Content of the Paper with Invisible Markup

```
21  % 5. Annotate the research problem, objective, method, result, and conclusion of the
    paper using the commands\researchproblem{}, \objective{}, \method{}, \result{},
    \conclusion{}
22  \begin{abstract}
23  [Background.] Empirical research in requirements engineering (RE) is a constantly
    evolving topic, with a growing number of publications. Several papers address this
    topic using literature reviews to provide a snapshot of its ``current'' state and
    evolution. However, these papers have never built on or updated earlier ones, resulting
    in overlap and redundancy. The underlying problem is the
    \researchproblem{unavailability of data from earlier works}. Researchers need technical
    infrastructures to conduct sustainable literature reviews. [Aims.] We examine the
    \objective{use of the Open Research Knowledge Graph (ORKG) as such an infrastructure to
    build and publish an initial Knowledge Graph of Empirical research in RE (KG-EmpiRE)
    whose data is openly available}. Our long-term goal is to continuously maintain KG-
    EmpiRE with the research community to synthesize a comprehensive, up-to-date, and long-
    term available overview of the state and evolution of empirical research in RE.
    [Method.] We conduct a \method{literature review using the ORKG} to build and publish
    KG-EmpiRE which we \method{evaluate against competency questions} derived from a
    published vision of empirical research in software (requirements) engineering for 2020
    -- 2025. [Results.] \result{From 570 papers of the IEEE International Requirements
    Engineering Conference (2000 -- 2022), we extract and analyze data on the reported
    empirical research} and \result{answer 16 out of 77 competency questions}. These
    answers show a positive development towards the vision, but also the need for future
    improvements. [Conclusions.] \conclusion{The ORKG is a ready-to-use and advanced
    infrastructure to organize data from literature reviews as knowledge graphs}. The
    resulting knowledge graphs make the data openly available and maintainable by research
    communities, enabling sustainable literature reviews.
24  \end{abstract}
```

What can I do, if the text is not suitable for annotation?

Example:

"… \method{evaluate against competency questions} …"

Solution:

\method***{evaluation against competency questions}

Remark:
This text is added to the PDF metadata, but **not rendered** in the text of the PDF.

6. Annotate Content of the Paper with Invisible Markup: Result

```
% 5. Annotate the research problem, objective, method, result, and conclusion of the paper using the commands\researchproblem{}, \objective{}, \method{}, \result{}, \conclusion{}
\begin{abstract}
[Background.] Empirical research in requirements engineering (RE) is a constantly evolving topic, with a growing number of publications. Several papers address this topic using literature reviews to provide a snapshot of its ``current'' state and evolution. However, these papers have never built on or updated earlier ones, resulting in overlap and redundancy. The underlying problem is the unavailability of data from earlier works. Researchers need technical infrastructures to conduct sustainable literature reviews. [Aims.] We examine the \objective{use of the Open Research Knowledge Graph (ORKG) as such an infrastructure to build and publish an initial Knowledge Graph of Empirical research in RE (KG-EmpiRE) whose data is openly available}. Our long-term goal is to continuously maintain KG-EmpiRE with the research community to synthesize a comprehensive, up-to-date, and long-term available overview of the state and evolution of empirical research in RE. [Method.] We conduct a \method{literature review using the ORKG} to build and publish KG-EmpiRE which we evaluate against competency questions derived from a published vision of empirical research in software (requirements) engineering for 2020 -- 2025. [Results.] \result{From 570 papers of the IEEE International Requirements Engineering Conference (2000 -- 2022), we extract and analyze data on the reported empirical research} and \result{answer 16 out of 77 competency questions}. These answers show a positive development towards the vision, but also the need for future improvements. [Conclusions.] \conclusion{The ORKG is a ready-to-use and advanced infrastructure to organize data from literature reviews as knowledge graphs}. The resulting knowledge graphs make the data openly available and maintainable by research communities, enabling sustainable literature reviews.
\end{abstract}

% 6. If written text is not suitable for annotation, we can also annotate invisible text using the *-notation
\researchproblem*{unavailability of the extracted and analyzed data from literature reviews}
\method*{evaluation against competency questions}
```

What can I do, if the text is not suitable for annotation?

Example:

"… \method{evaluate against competency questions} …"

Solution:

\method***{evaluation against competency questions}

Remark:
This text is added to the PDF metadata, but **not rendered** in the text of the PDF.

Comparison of Annotated Paper Versions

```
21  % 5. Annotate the research problem, objective, method, result, and conclusion of the
    paper using the commands\researchproblem{}, \objective{}, \method{}, \result{},
    \conclusion{}
22  \begin{abstract}
23  [Background.] Empirical research in requirements engineering (RE) is a constantly
    evolving topic, with a growing number of publications. Several papers address this
    topic using literature reviews to provide a snapshot of its ``current'' state and
    evolution. However, these papers have never built on or updated earlier ones, resulting
    in overlap and redundancy. The underlying problem is the
    \researchproblem{unavailability of data from earlier works}. Researchers need technical
    infrastructures to conduct sustainable literature reviews. [Aims.] We examine the
    \objective{use of the Open Research Knowledge Graph (ORKG) as such an infrastructure to
    build and publish an initial Knowledge Graph of Empirical research in RE (KG-EmpiRE)
    whose data is openly available}. Our long-term goal is to continuously maintain KG-
    EmpiRE with the research community to synthesize a comprehensive, up-to-date, and long-
    term available overview of the state and evolution of empirical research in RE.
    [Method.] We conduct a \method{literature review using the ORKG} to build and publish
    KG-EmpiRE which we \method{evaluate against competency questions} derived from a
    published vision of empirical research in software (requirements) engineering for 2020
    -- 2025. [Results.] \result{From 570 papers of the IEEE International Requirements
    Engineering Conference (2000 -- 2022), we extract and analyze data on the reported
    empirical research} and \result{answer 16 out of 77 competency questions}. These
    answers show a positive development towards the vision, but also the need for future
    improvements. [Conclusions.] \conclusion{The ORKG is a ready-to-use and advanced
    infrastructure to organize data from literature reviews as knowledge graphs}. The
    resulting knowledge graphs make the data openly available and maintainable by research
    communities, enabling sustainable literature reviews.
24  \end{abstract}
```

```
21  % 5. Annotate the research problem, objective, method, result, and conclusion of the
    paper using the commands\researchproblem{}, \objective{}, \method{}, \result{},
    \conclusion{}
22  \begin{abstract}
23  [Background.] Empirical research in requirements engineering (RE) is a constantly
    evolving topic, with a growing number of publications. Several papers address this
    topic using literature reviews to provide a snapshot of its ``current'' state and
    evolution. However, these papers have never built on or updated earlier ones, resulting
    in overlap and redundancy. The underlying problem is the unavailability of data from
    earlier works. Researchers need technical infrastructures to conduct sustainable
    literature reviews. [Aims.] We examine the \objective{use of the Open Research
    Knowledge Graph (ORKG) as such an infrastructure to build and publish an initial
    Knowledge Graph of Empirical research in RE (KG-EmpiRE) whose data is openly
    available}. Our long-term goal is to continuously maintain KG-EmpiRE with the research
    community to synthesize a comprehensive, up-to-date, and long-term available overview
    of the state and evolution of empirical research in RE. [Method.] We conduct a
    \method{literature review using the ORKG} to build and publish KG-EmpiRE which we
    evaluate against competency questions derived from a published vision of empirical
    research in software (requirements) engineering for 2020 -- 2025. [Results.]
    \result{From 570 papers of the IEEE International Requirements Engineering Conference
    (2000 -- 2022), we extract and analyze data on the reported empirical research} and
    \result{answer 16 out of 77 competency questions}. These answers show a positive
    development towards the vision, but also the need for future improvements.
    [Conclusions.] \conclusion{The ORKG is a ready-to-use and advanced infrastructure to
    organize data from literature reviews as knowledge graphs}. The resulting knowledge
    graphs make the data openly available and maintainable by research communities,
    enabling sustainable literature reviews.
24  \end{abstract}
25
26  % 6. If written text is not suitable for annotation, we can also annotate invisible
    text using the *-notation
27  \researchproblem*{unavailability of the extracted and analyzed data from literature
    reviews}
28  \method*{evaluation against competency questions}
```

7. (Optional) Annotate Research Field

Remark:

1. A paper in the ORKG is assigned to a research field based on the DFG classification. All research fields: https://orkg.org/fields.
2. URI of the ORKG semantic web resource for Software Engineering: https://orkg.org/resource/R140.
3. Use the command \uri{“**URI**”}{“**Label**”} inside an annotation to refer to resources in the semantic web. The first argument is the **URI** to the semantic resource and the second is an optional **Label**.
4. The term “Software Engineering" does not appear in the abstract, so we need an **invisible** annotation and we also use the \uri{}{} command to create a reference to the semantic resource.

1. Annotate the research field

\researchfield{}

2. Refer to the ORKG semantic web resource Software Engineering

\uri{“URI”}{“Label”}

7. (Optional) Annotate Research Field: Result

```
29  % 7. Optional: Annotate research field of the paper
30  \researchfield*{\uri{https://orkg.org/resource/R140}{Software Engineering}}
```

Remark:

1. A paper in the ORKG is assigned to a research field based on the DFG classification. All research fields: https://orkg.org/fields.
2. URI of the ORKG semantic web resource for Software Engineering: https://orkg.org/resource/R140.
3. Use the command \uri{“**URI**”}{“**Label**”} inside an annotation to refer to resources in the semantic web. The first argument is the **URI** to the semantic resource and the second is an optional **Label**.
4. The term “Software Engineering" does not appear in the abstract, so we need an **invisible** annotation and we also use the \uri{}{} command to create a reference to the semantic resource.

1. Annotate the research field

\researchfield{}

2. Refer to the ORKG semantic web resource Software Engineering

\uri{“URI”}{“Label”}

8. Use Custom Annotations of REFSQ'24 and REFSQ'25

```
33  % 8. Optional: Using REFSQ'24 and REFSQ'25 annotations
34  \contribution*{code repository}{\url{https://github.com/okarras/EmpiRE-Analysis}}
35  \contribution*{dataset}{\url{https://orkg.org/api/rdf/dump}}
36
37  \end{document}
```

Remark:

1. Use the command \contribution{"**Property name**"}{"**Label**"} to add a custom annotation for your domain. The first argument is the **Property name** of the property from the ORKG you want to use, and the second is an optional **Label**. SciKGTeX checks if a property with the provided exact **Property name** exists and replaces it with the internal property ID in the ORKG namespace.
2. All ORKG properties can be found here: https://orkg.org/properties.

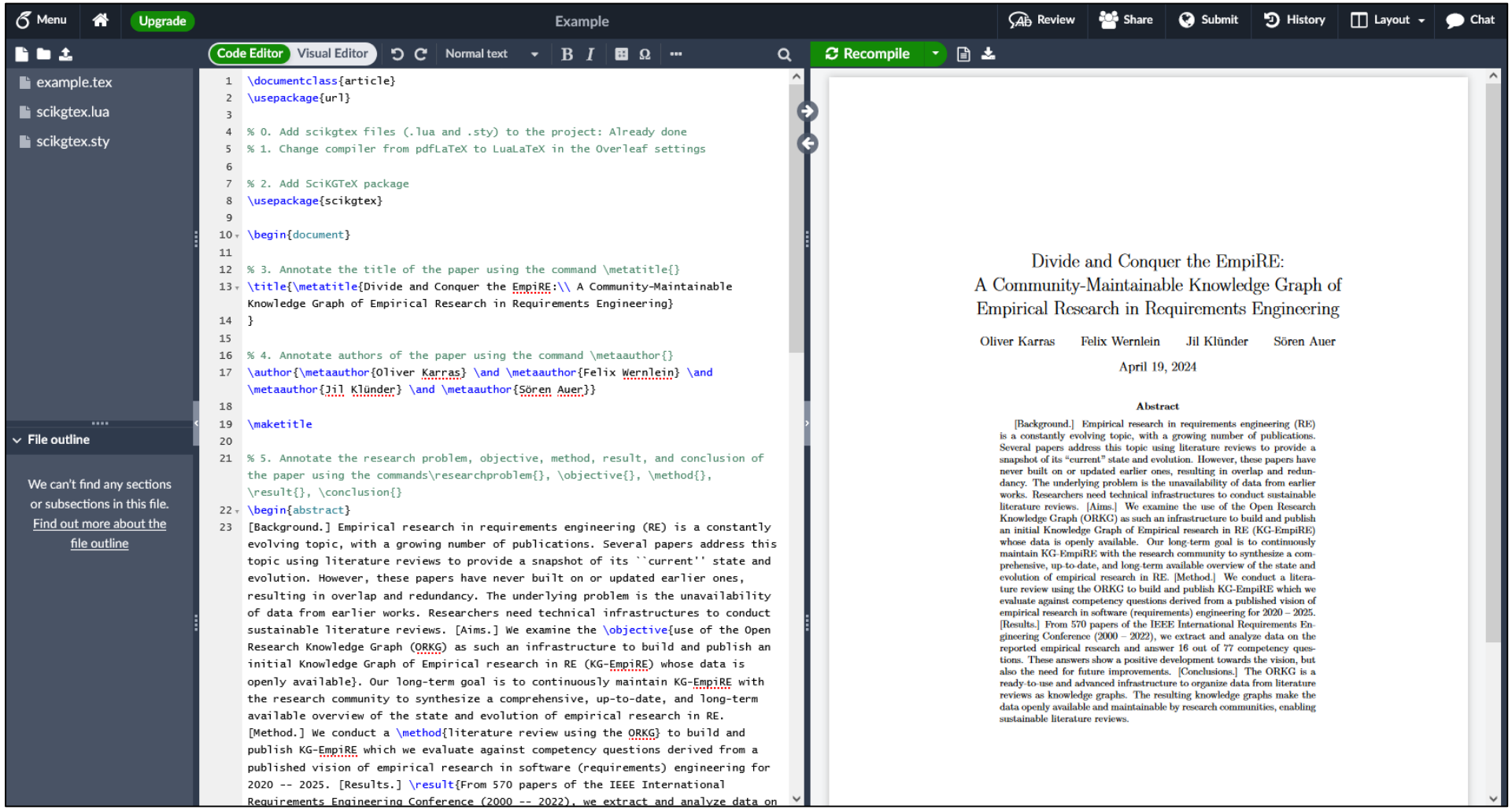
1. Annotate the code repository

\contribution{code repository}{}

2. Annotate the dataset

\contribution{dataset}{}

9. Generate FAIR-Annotated PDF of the Paper

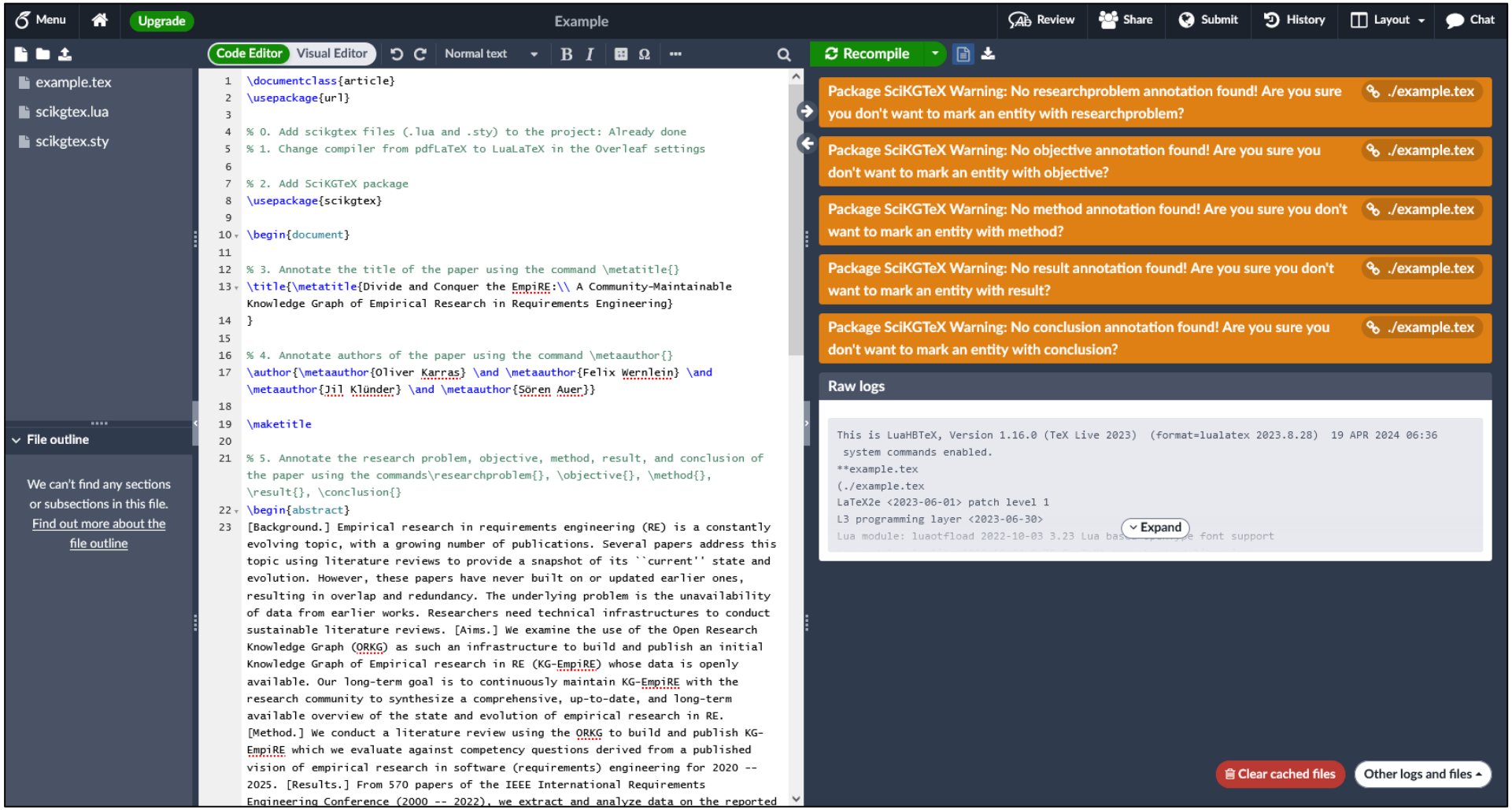


Simply…

recompile

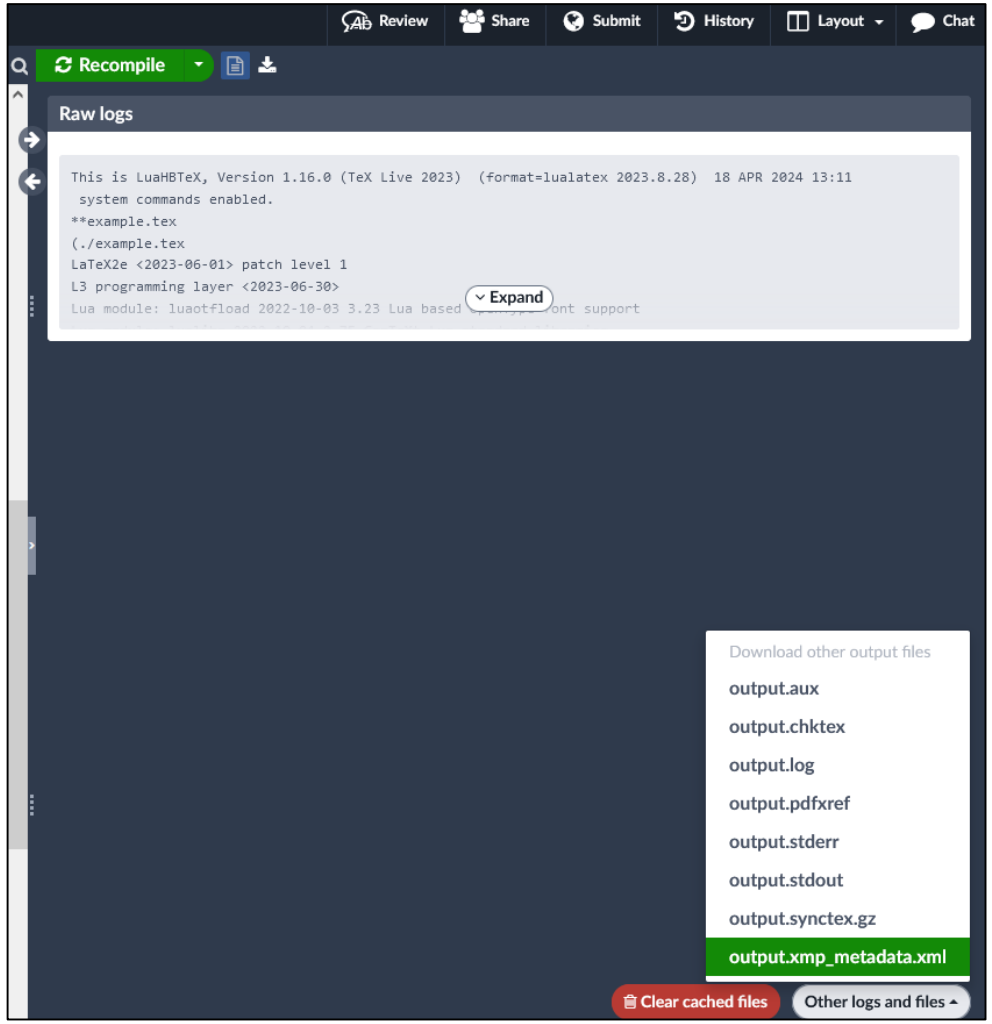
Remark:
Each recompile adds the annotations to the metadata of the generated PDF.

9. Generate FAIR-Annotated PDF of the Paper



For the 5 predefined commands for content, SciKGTeX also provides **warnings** if an annotation is missing.

10. Check the FAIR information embedded in the PDF



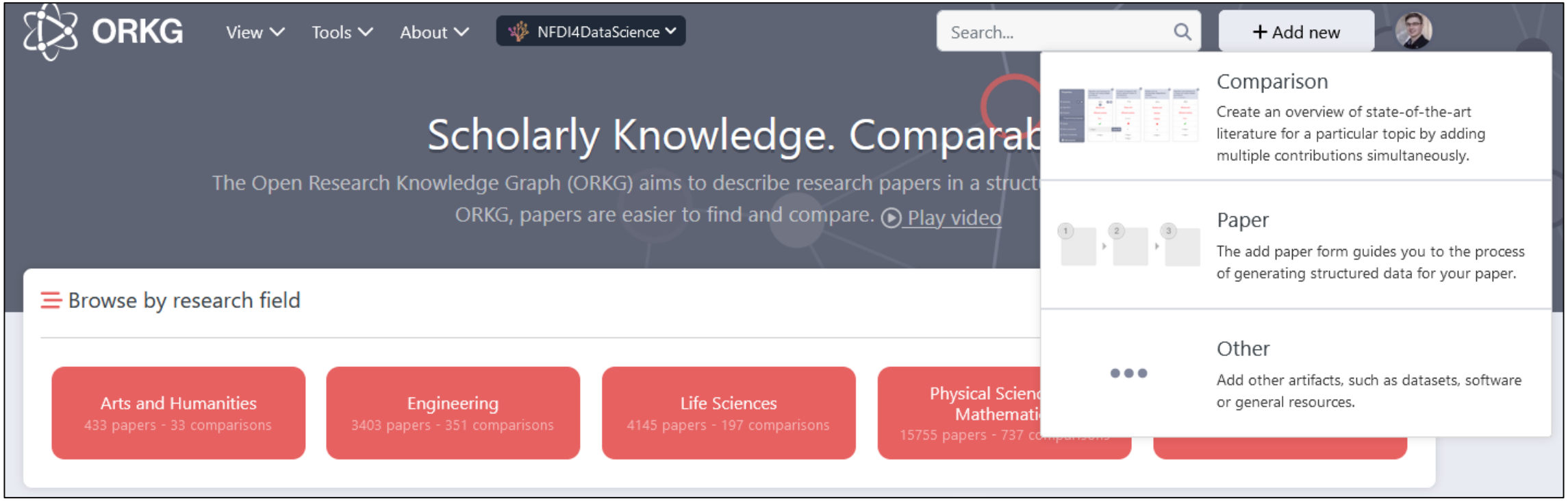
1. Open “Logs and outputs files”
2. Select “Other logs and files”
3. Download “output.xmp_metadata.xml”
4. Open “output.xmp_metadata.xml”

Remark:
We only use an abstract for the annotations as a simplified example. Annotations can be used **anywhere** (in one or more LaTeX files).

10. Check the FAIR information embedded in the PDF


```
<?xpacket begin="?" id="a7c48312-233b-400b-c0f3-0c296941c6"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/">
 <rdf:RDF xmlns:orkg="http://orkg.org/core#" xmlns:orkg_property="http://orkg.org/property/" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
   <rdf:Description rdf:about="https://www.orkg.org/orkg/paper/a7c48312-233b-400b-c0f3-0c296941c6">
     <rdf:type rdf:resource="http://orkg.org/core#Paper"/>
     <orkg:hasTitle>Divide and Conquer the EmpiRE: A Community-Maintainable Knowledge Graph of Empirical Research in Requirements
     Engineering</orkg:hasTitle>
     <orkg:hasAuthor>Oliver Karras</orkg:hasAuthor>
     <orkg:hasAuthor>Felix Wernlein</orkg:hasAuthor>
     <orkg:hasAuthor>Jil Klünder</orkg:hasAuthor>
     <orkg:hasAuthor>Sören Auer</orkg:hasAuthor>
    <orkg_property:P30>
     <rdf:Description rdf:about="https://orkg.org/resource/R140">
       <rdfs:label>Software Engineering</rdfs:label>
      </rdf:Description>
     </orkg_property:P30>
    <orkg:hasResearchContribution>
     <orkg:ResearchContribution rdf:about="https://www.orkg.org/orkg/paper/a7c48312-233b-400b-c0f3-0c296941c6/contribution_ORKG_default">
       <orkg_property:P15051>use of the Open Research Knowledge Graph (ORKG) as such an infrastructure to build and publish an initial Knowledge
       Graph of Empirical research in RE (KG-EmpiRE) whose data is openly available</orkg_property:P15051>
       <orkg_property:P1005>literature review using the ORKG</orkg_property:P1005>
       <orkg_property:P1006>From 570 papers of the IEEE International Requirements Engineering Conference (2000 -- 2022), we extract and analyze
       data on the reported empirical research</orkg_property:P1006>
       <orkg_property:P1006>answer 16 out of 77 competency questions</orkg_property:P1006>
       <orkg_property:P15419>The ORKG is a ready-to-use and advanced infrastructure to organize data from literature reviews as knowledge
       graphs</orkg_property:P15419>
       <orkg_property:P32>unavailability of the extracted and analyzed data from literature reviews</orkg_property:P32>
       <orkg_property:P1005>evaluation against competency questions</orkg_property:P1005>
       <orkg_property:P49000>https://github.com/okarras/EmpiRE-Analysis</orkg_property:P49000>
       <orkg_property:P2005>https://orkg.org/api/rdf/dump</orkg_property:P2005>
      </orkg:ResearchContribution>
     </orkg:hasResearchContribution>
    </rdf:Description>
  </rdf:RDF>
</x:xmpmeta>
<?xpacket end="r"?>
```

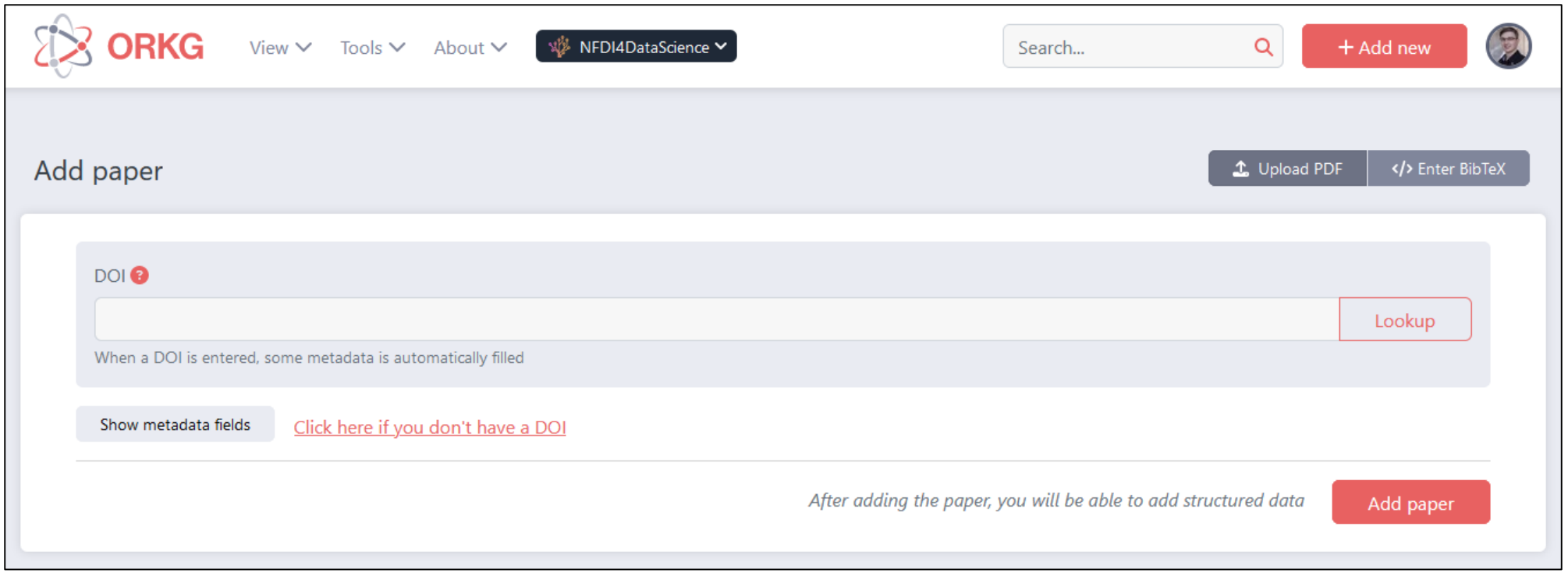
11. Import the FAIR information into the ORKG



1. Log in with your credentials
2. Click “+ Add new”
3. Select “Paper”

Remark:
ORKG does **not save** the uploaded PDF.
It **only extracts** the FAIR annotations embedded in the PDF metadata.

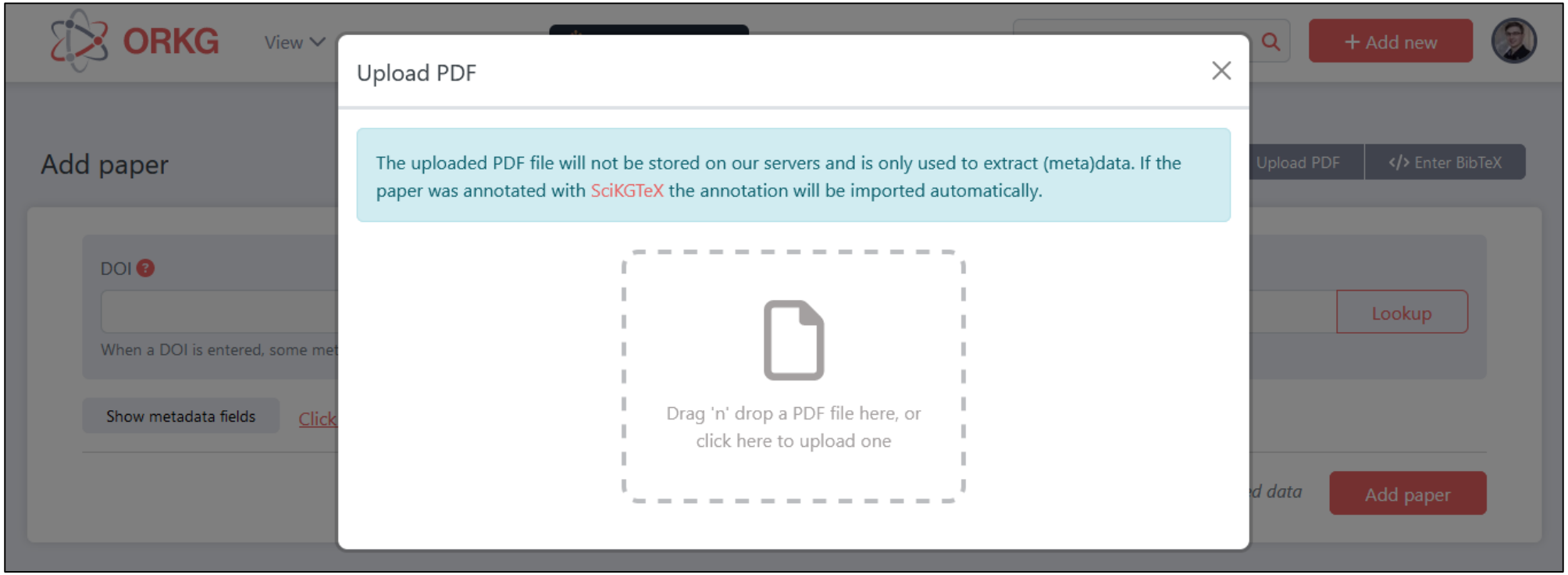
11. Import the FAIR information into the ORKG



1. Select “Upload PDF”

Remark:
ORKG does **not save** the uploaded PDF. It **only extracts** the FAIR annotations embedded in the PDF metadata.

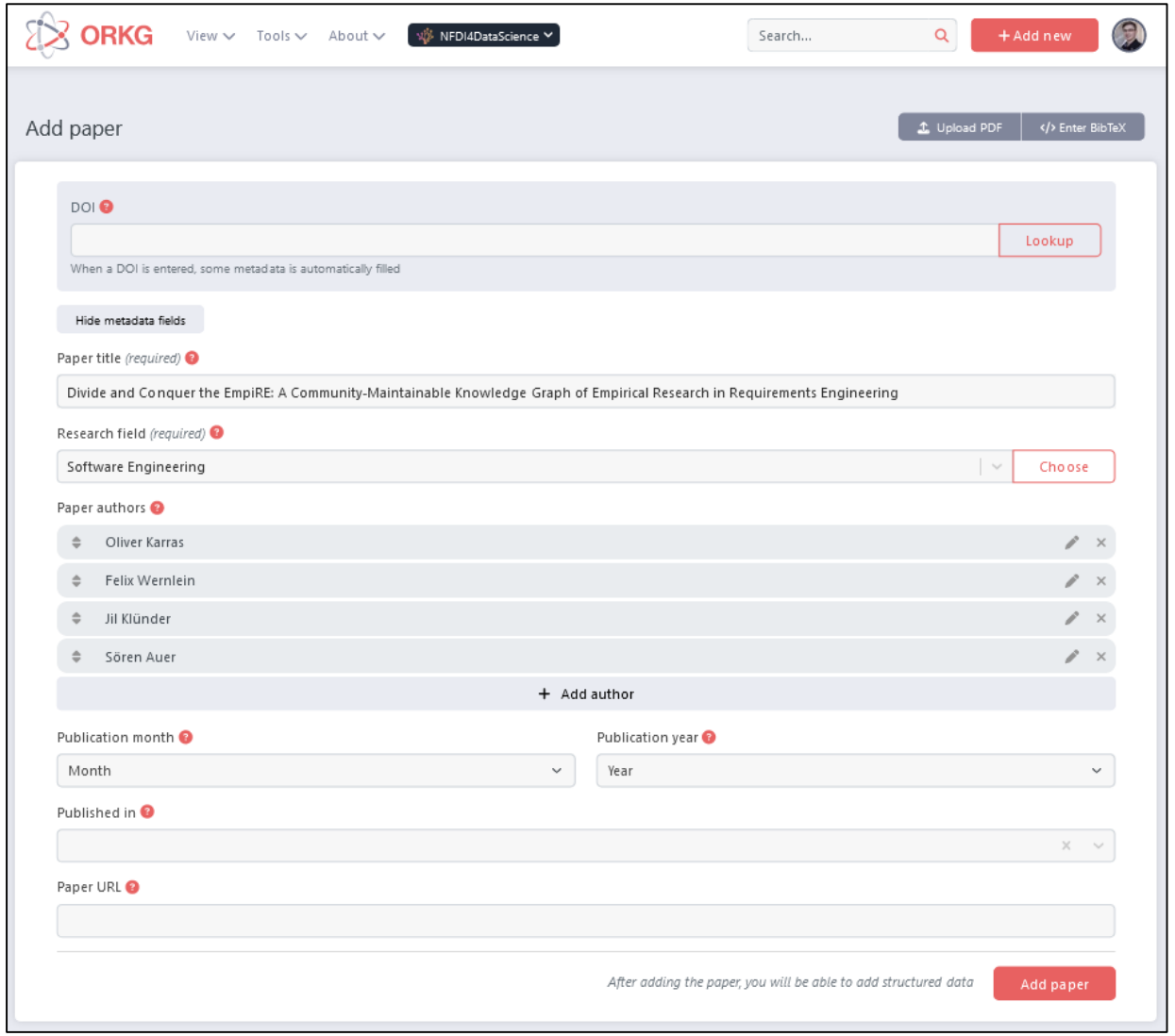
11. Import the FAIR information into the ORKG



1. Select or drag your PDF file

Remark:
ORKG does **not save** the uploaded PDF. It **only extracts** the FAIR annotations embedded in the PDF metadata.

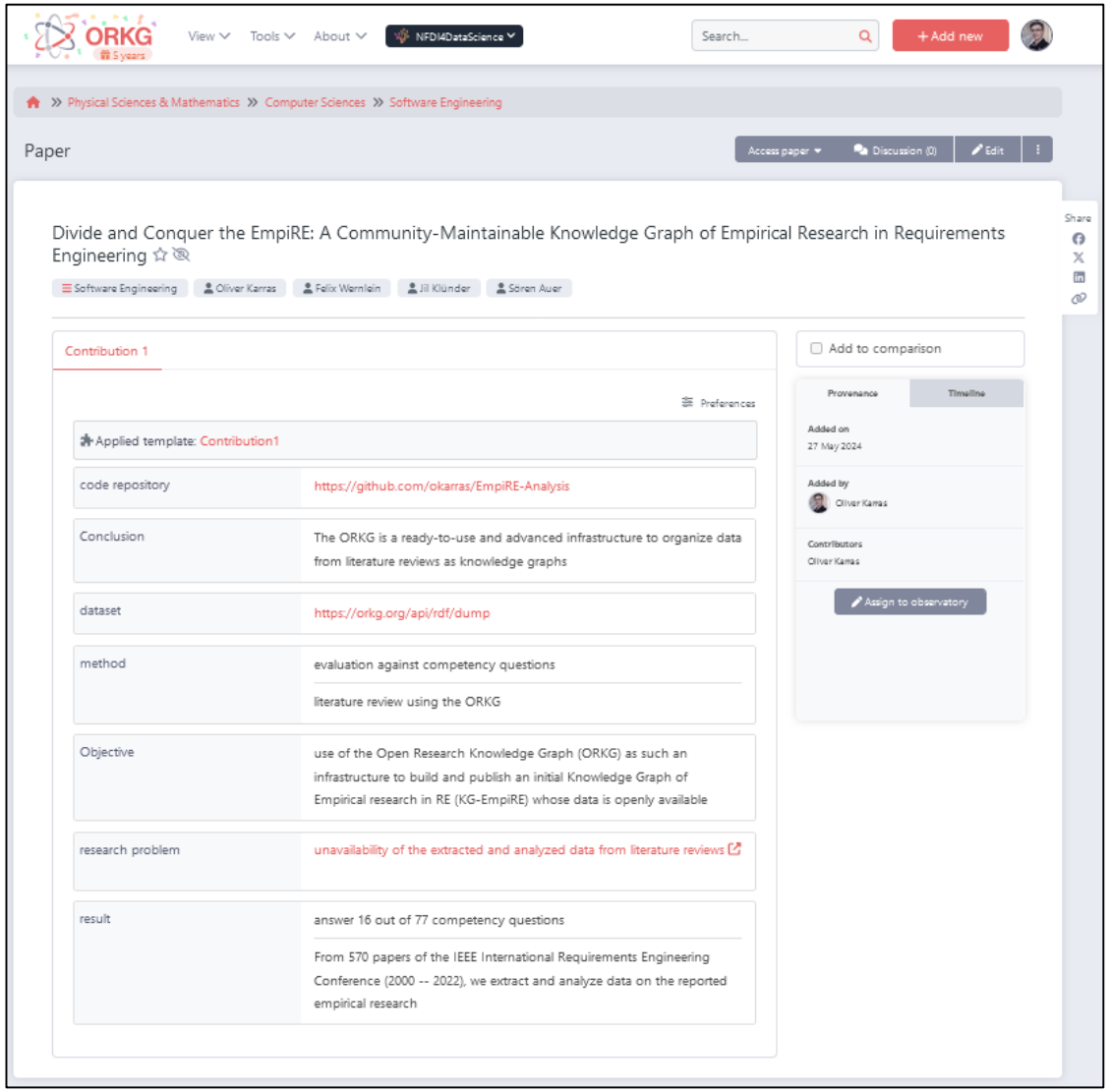
11. Import the FAIR information into the ORKG

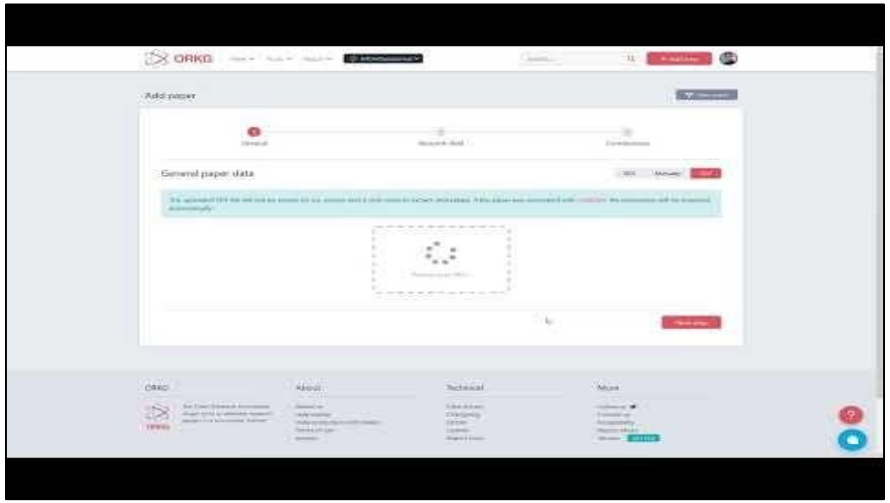


1. ORKG shows imported metadata
2. Optional: Edit the metadata
3. Select “Add paper” at the bottom

Remark:
ORKG does **not save** the uploaded PDF. It **only extracts** the FAIR annotations embedded in the PDF metadata.

11. Import the FAIR information into the ORKG: Result





https://www.youtube.com/watch?v=ZzrQ_YCKVsYa

Remark:
ORKG does **not save** the uploaded PDF.
It **only extracts** the FAIR annotations embedded in the PDF metadata.

ORKG: Describing Papers Manually and Creating an ORKG Comparison

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

June 24th, 2024, Reykjavik, Iceland

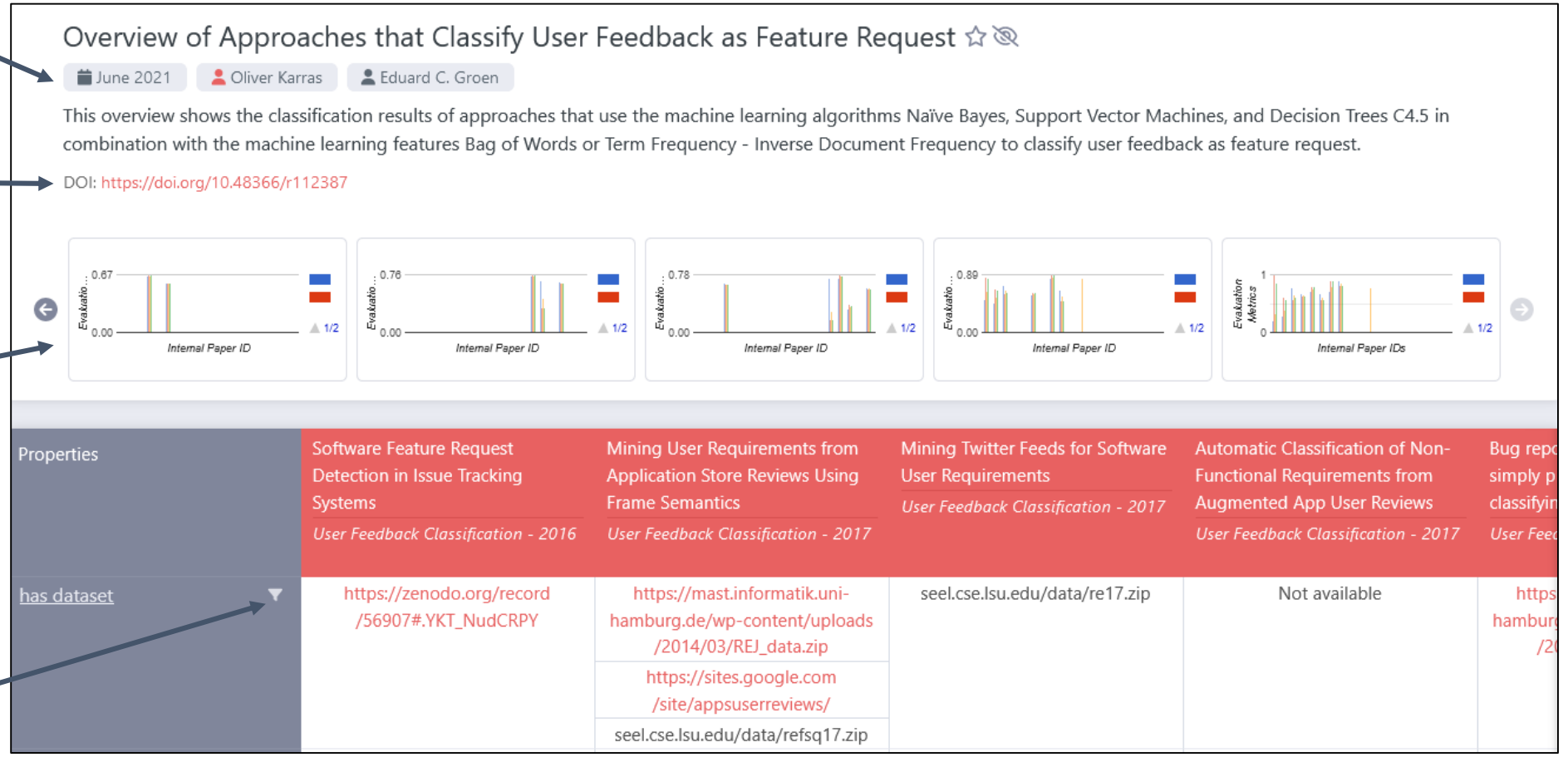
ORKG Comparisons

Acknowledgement of creators

DOI

Visualizations

Interactive filtering



Example: ORKG Comparison of Related Work – State of the Art[5]

A Comparison of Scientific Publications on the State of Empirical Research in Requirements Engineering and Software Engineering

November 2023 | Oliver Karras | Felix Wernlein | Jil Ann-Christin Klünder | Sören Auer

Empirical Software Engineering

This comparison provides an overview of scientific publications that have investigated primary studies in requirements engineering and software engineering to give a snapshot of the "current" state of empirical research in requirements engineering and software engineering. In particular, the comparison shows for each publication (1) which research fields and topics were investigated, (2) whether and where the extracted and analyzed data is available, and (3) which method was used to determine the state, including further details about the respective method.

DOI: https://doi.org/10.48366/R650023

Properties	Empirical research in requirements engineering: trends and opportunities *Empirical research - 2016*	Empirical research methodologies and studies in Requirements Engineering: How far did we come? *Empirical research - 2014*	A Survey on Empirical Requirements Engineering Research Practices *Empirical research - 2012*	Evidence-Based Structuring and Evaluation of Empirical Research in Requirements Engineering: Fundamentals, Framework, Research Map *Empirical research - 2010*	An Analy... Requirer... Data *Empirica...*
research problem	empirical research in requirements engineering	empirical research in requirements engineering	empirical research in requirements engineering	empirical research in requirements engineering	empirica...
research field investigated	Requirements Engineering	Requirements Engineering	Requirements Engineering	Requirements Engineering	Requ...
topic investigated	bibliographic metadata context data collection	bibliographic metadata research topic theory	context	context	

https://doi.org/10.48366/R650023

Figure 1: *Comparison* of related publications on the "current" state and evolution of empirical research in RE and SE [16].

Figure 1 shows an excerpt from a comparison that we created to get an overview of related publications on the "current" state and evolution of empirical research in RE and SE [16]. For three publications, the excerpt shows the method used, the data availability, as well as the period and the number of papers examined. We use the ORKG due to its cross-domain and cross-topic characteristics, as well as its successful application for CrowdRE by Karras et al. [42].

III. Related Work

Below, we review 14 publications that provide snapshots of the "current" state and evolution of empirical research in RE and SE (see Table I) [16]. We only consider publications that address the topic in general and are not limited to specific aspects, such as a method [67], [68] or a context [69], [70].

We found five publications on empirical research in RE published between 2005 and 2016 and nine on empirical research in SE published between 2002 and 2021. While one publication [1] examined empirical research in RE using a survey with 42 respondents, the other 13 publications [2], [3], [5]–[15] used (systematic) literature reviews or systematic mapping studies to analyze on average 402.9 papers (minimum: 20, median: 154, and maximum: 2237 papers) published between 1977 and 2019 with overlapping periods. In total, these 13 publications examined papers from a total of 60 different venues on 18 different themes. Nine of the 60 venues and ten of the 18 themes were examined by more than two publications. These facts show that there is considerable overlap and redundancy between these publications in terms of goals, methods used, periods, venues, and themes examined. This overlap and redundancy could have been avoided if researchers had collaborated to build on and update earlier works. However, only four out of 14 publications offer their data at all, with only one publication [15] using a public data repository [71], [72]. The other three publications only offer links that no longer work [5], [12], [13].

In terms of key findings, the 14 publications show consistent results, although not all 18 themes were examined in all publications. For example, eleven of the 14 publications reported on the most commonly used research methods. Until 2000, the most common research methods were conceptual analysis and concept implementation [7]. Between 2000 and 2015, the most commonly used research methods changed to case studies and experiments [3], [5], [8]–[11], which were expanded after 2015 to also include surveys and systematic literature reviews [12]–[15]. While this change shows an evolution of research methods used, we also note that experiments and case studies have been the two main research methods for empirical research in RE and SE for more than 20 years. Although these two research methods have been used for a long time, seven publications concluded that there is a need to develop, expand, and use standardized terminology and theories (from other disciplines) to more consistently represent the empirical research conducted and better explain the results found [1], [2], [6]–[8], [12], [13]. In this regard, seven publications also analyzed the information reported for a comprehensive description of a research design. This information includes details about the research question(s) [1], contextual factors [6],

Table I: Details of related publications on the "current" state and evolution of empirical research in RE and SE [16].
Legend: Literature Review (LR), Systematic Literature Review (SLR), and Systematic Mapping Study (SMS)

Paper	Year	Field	Method	Period	Data basis	Dataset	Venues (Frequency > 2)	Themes (Frequency > 2)
[6]	2005	RE	LR	1986 – 2002	35 papers	Unavailable	1) Empirical Software Engineering Journal (8) 2) IEEE Software (4) 3) Requirements Engineering Journal (4) 4) ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (4) 5) IEEE Transactions on Software Engineering (3) 6) Information and Software Technology Journal (3) 7) IEEE International Requirements Engineering Conference (3) 8) Journal of Systems and Software (3) 9) International Conference on Software Engineering (3)	1) Data collection (12) 2) Research method (11) 3) Bibliographic metadata (10) 4) Data analysis (8) 5) Research paradigm (7) 6) Research design (7) 7) Research topic (5) 8) Research context (4) 9) Sample of population (4) 10) Theory (3)
[7]	2010		SLR	Unknown	154 papers	Broken link		
[1]	2012		Survey	10.03.2012 – 30.03.2012	42 respondents	Unavailable		
[2]	2014		LR	1983 – 2013	2237 papers	Unavailable		
[3]	2016		SMS	Open – 2012	270 papers	Unavailable		
[7]	2002	SE	LR	1995 – 1999	369 papers	Unavailable		
[8]	2002		LR	1996 – 2002	68 papers	Unavailable		
[9]	2005		LR	1997 – 2005	119 papers	Unavailable		
[10]	2006		LR	1977 – 2005	63 papers	Unavailable		
[11]	2007		LR	1996 – 2005	133 papers	Unavailable		
[12]	2015		SMS	1996 – 2013	891 papers	Broken link		
[13]	2018		SMS	2013 – 2017	538 papers	Broken link		
[14]	2019		SMS	1993 – 2014	541 papers	Unavailable		
[15]	2021		SMS	Open – 2019	20 papers	Available		

[5] Karras et al.: *Divide and Conquer the EmpiRE: A Community-Maintainable Knowledge Graph of Empirical Research in Requirements Engineering.* 2023 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), DOI: 10.1109/ESEM56168.2023.10304795, 2023.

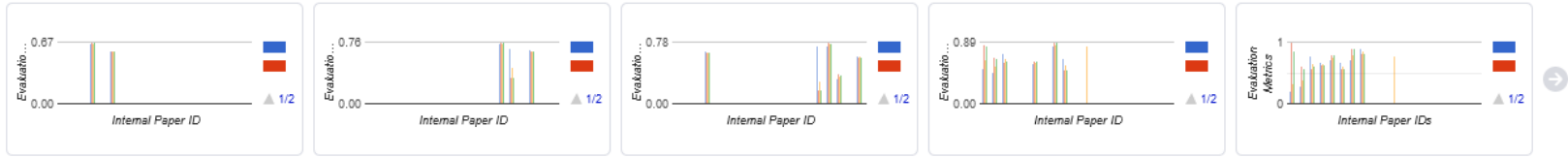
Example: ORKG Comparison of Literature – ML Approaches[9]

Overview of Approaches that Classify User Feedback as Feature Request

June 2021 | Oliver Karras | Eduard C. Groen

This overview shows the classification results of approaches that use the machine learning algorithms Naïve Bayes, Support Vector Machines, and Decision Trees C4.5 in combination with the machine learning features Bag of Words or Term Frequency - Inverse Document Frequency to classify user feedback as feature request.

DOI: https://doi.org/10.48366/r112387



Properties	Software Feature Request Detection in Issue Tracking Systems *User Feedback Classification - 2016*	Mining User Requirements from Application Store Reviews Using Frame Semantics *User Feedback Classification - 2017*	Mining Twitter Feeds for Software User Requirements *User Feedback Classification - 2017*	Automatic Classification of Non-Functional Requirements from Augmented App User Reviews *User Feedback Classification - 2017*	Bug repo simply p classifyin *User Feed*
has dataset	https://zenodo.org/record/56907#.YKT_NudCRPY	https://mast.informatik.uni-hamburg.de/wp-content/uploads/2014/03/REJ_data.zip https://sites.google.com/site/appsuserreviews/ seel.cse.lsu.edu/data/refsq17.zip	seel.cse.lsu.edu/data/re17.zip	Not available	https hamburg /2

https://doi.org/10.48366/r112387

Visualizations

Fig. 2: Excerpt from our comparison for Case II [35].

77 papers from the SLR by Khan et al. [18], describing the relation of the papers to five phases of RE and the CrowdRE utilities applied [36]. We are still in the process of adding the contributions from the remaining 50 papers, which is more time-consuming than for the quantitative data from Case I because of the expert judgments needed for classifying the papers' contributions. The comparison of the 27 papers makes it easy to identify, for example, the four papers that address the runtime purpose of monitoring for requirements evolution.

With the created comparisons [35], [36], we achieved our goal of acquiring and curating the detailed results of both SLRs. The knowledge-based representation in the form of comparisons has several advantages over a purely document-based representation. The comparisons are interactive and allow filtering of views by different scholarly knowledge contained in each row, even by specific value ranges of qualitative and quantitative content. The ORKG also provides a service for generating several graphical visualizations based on data in the comparisons, helping the reader understand information faster than through the large comparison table. The most important feature of the ORKG is that the added contributions and created comparisons are available to anyone. In this way, every crowd member can use the curated scholarly knowledge and created comparisons as a basis for new comparisons. Moreover, the existing comparisons can be expanded with additional scholarly knowledge from papers already included, and even with new contributions from papers added later to the ORKG. We already expanded several contributions, e.g., the results of other classifications reported in Dhinakaran et al.'s paper [40][10]. For Case I, we added the details of the three crowd properties *scale*, *level of knowledge*, *skills & expertise*; and *roles*, which are only briefly and superficially described in the SLR [18]. For Case II, we added links to the datasets used and performance values to classification categories other than "Feature Request". This expansion is relevant to enable long-term curation. For example, a development succeeding the SLR by Santos et al. [17] are reports of Deep Learning algorithms showing promising results in classifying user feedback [41], [42], which should be successively added to the comparison.

Despite all these advantages, the ORKG also has limitations. Most of the limitations we experienced can be attributed to the development status of the platform, which is currently in beta. Further development of the ORKG must improve interactions for the expert crowd by enabling better workflows for entering data and creating visualizations. Nevertheless, we also experienced that the project team has always responded directly to our reported issues, which we could see getting added to the GitLab issue tracker[11] and addressed shortly thereafter.

V. Discussion

The ORKG aroused our interest as a crowdsourcing platform for applying and communicating CrowdRE research. In this experience report, we explored whether the ORKG can promote the potential of CrowdRE in open source and open research settings, taking two perspectives: that of CrowdRE researchers and that of crowd members.

Our first contribution is that we provide a comprehensive overview of the ORKG's features as a crowdsourcing platform for acquiring and curating scholarly knowledge [37], mapped to the four key activities of CrowdRE. Our findings show that the ORKG is a crowdsourcing platform offering several features that can facilitate successful CrowdRE. Although the ORKG project team has not yet consciously applied CrowdRE, they already address crucial parts of the CrowdRE cycle by motivating crowd members to participate, eliciting feedback, and monitoring context & usage data, which they analyze to derive and implement the needs and requirements of the crowd.

To motivate crowd members, the project team uses established mechanisms and incentives to boost intrinsic and extrinsic motivation (see Finding 1). Feedback is elicited

[10] https://www.orkg.org/orkg/paper/R76818/R76825

[11] https://gitlab.com/TIBHannover/orkg/orkg-frontend/-/issues/634

[9] Karras et al.: *Researcher or Crowd Member? Why not both! The Open Research Knowledge Graph for Applying and Communicating CrowdRE Research.* 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW), DOI: 10.1109/REW53955.2021.00056, 2021.

Example: ORKG Comparison of Literature – Simulation Parameters[10]

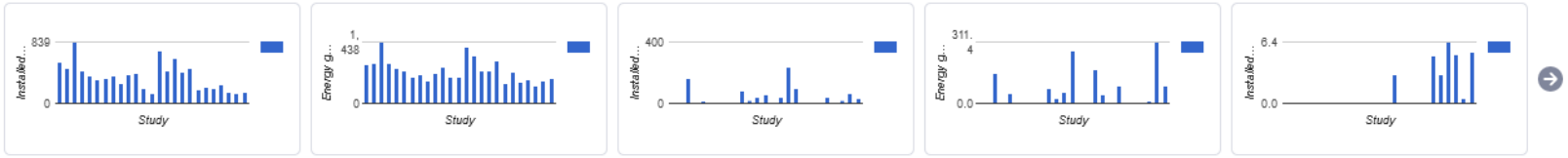
Comparison of Studies on Germany's Energy Supply in 2050

November 2021 | Felix Kullmann | Jan Göpfert | Oliver Karras | Patrick Kuckertz | Sören Auer | Markus Stocker | Peter Markewitz | Leander Kotzur | Detlef Stolten

NFDI4Ing Energy System Research

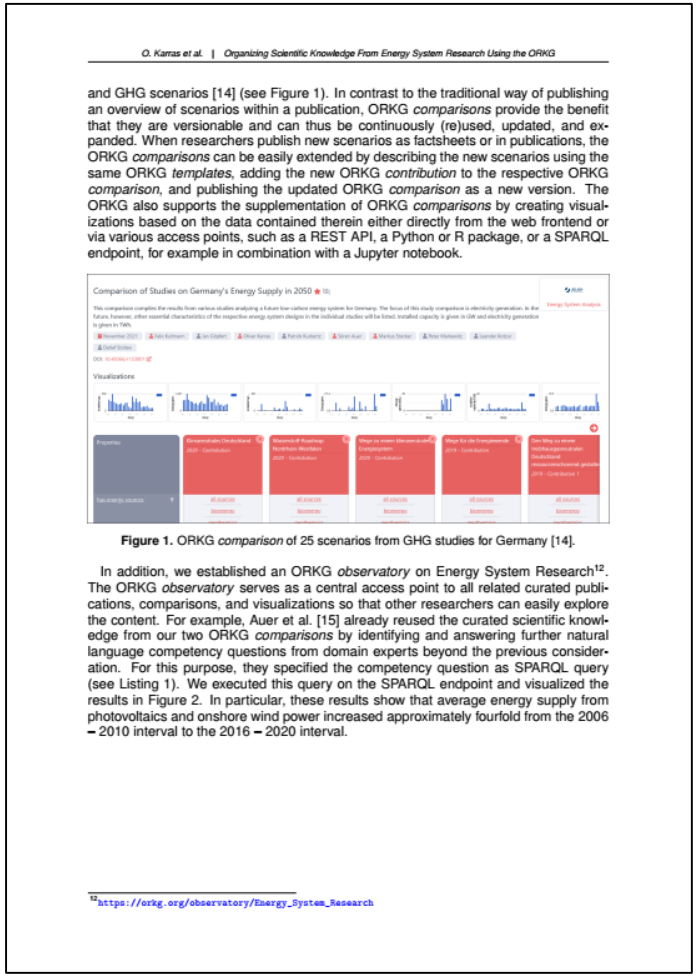
This comparison compiles the results from various studies analyzing a future low-carbon energy system for Germany. The focus of this study comparison is electricity generation. In the future, however, other essential characteristics of the respective energy system designs in the individual studies will be listed. Installed capacity is given in GW and electricity generation is given in TWh. The authors would like to thank the German Federal Government, the German State Governments, and the Joint Science Conference (GWK) for their funding and support as part of the NFDI4Ing consortium. Funded by the German Research Foundation (DFG) - project number: 442146713. This work was also supported by the Helmholtz Association under the program "Energy System Design".

DOI: https://doi.org/10.48366/r153801



Properties	Klimaneutrales Deutschland *Contribution - 2020*	Wasserstoff-Roadmap Nordrhein-Westfalen *Contribution - 2020*	Wege zu einem klimaneutralen Energiesystem *Contribution - 2020*	Wege für die Energiewende *Contribution - 2019*	Den We... treibhau... Deutsch... gestalte... *Contribu...*
has energy sources	all sources	all sources	all sources	all sources	
	bioenergy	bioenergy	bioenergy	bioenergy	
	geothermics	geothermics	geothermics	geothermics	
	hydropower	hydropower	hydropower	hydropower	
	import	import			
	net import	net import			

https://doi.org/10.48366/r153801



O. Karras et al. | Organizing Scientific Knowledge From Energy System Research Using the ORKG

and GHG scenarios [14] (see Figure 1). In contrast to the traditional way of publishing an overview of scenarios within a publication, ORKG *comparisons* provide the benefit that they are versionable and can thus be continuously (re)used, updated, and expanded. When researchers publish new scenarios as factsheets or in publications, the ORKG *comparisons* can be easily extended by describing the new scenarios using the same ORKG *templates*, adding the new ORKG *contribution* to the respective ORKG *comparison*, and publishing the updated ORKG *comparison* as a new version. The ORKG also supports the supplementation of ORKG *comparisons* by creating visualizations based on the data contained therein either directly from the web frontend or via various access points, such as a REST API, a Python or R package, or a SPARQL endpoint, for example in combination with a Jupyter notebook.

Figure 1. ORKG *comparison* of 25 scenarios from GHG studies for Germany [14].

In addition, we established an ORKG *observatory* on Energy System Research[12]. The ORKG *observatory* serves as a central access point to all related curated publications, comparisons, and visualizations so that other researchers can easily explore the content. For example, Auer et al. [15] already reused the curated scientific knowledge from our two ORKG *comparisons* by identifying and answering further natural language competency questions from domain experts beyond the previous consideration. For this purpose, they specified the competency question as SPARQL query (see Listing 1). We executed this query on the SPARQL endpoint and visualized the results in Figure 2. In particular, these results show that average energy supply from photovoltaics and onshore wind power increased approximately fourfold from the 2006 – 2010 interval to the 2016 – 2020 interval.

[12] https://orkg.org/observatory/Energy_System_Research

[10] Karras et al.: *Organizing Scientific Knowledge From Energy System Research Using the Open Research Knowledge Graph.* 1st NFDI4Energy Conference, DOI: 10.48550/arXiv.2401.13365, 2024.

Example: ORKG Comparison of Literature – Process Variants[11]

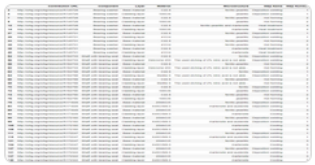
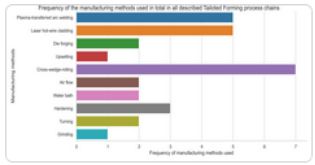
Tailored Forming Process Chain for the Manufacturing of Hybrid Components with Bearing Raceways Using Different Material Combinations ★

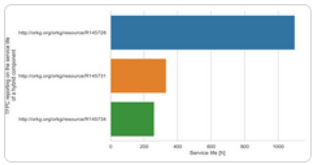
May 2022 | Oliver Karras | Laura Budde | Paulina Merkel

This comparison provides an overview of Tailored Forming process chains carried out in the CRC 1153 "Tailored Forming" for the manufacturing of hybrid components with bearing raceways using different material combinations. A variety of materials combinations is used for the Tailored Forming of such hybrid components. The comparison shows the combined materials, the entire Tailored Forming process chain with its individual steps, and the resulting hybrid component with its qualities. For each step, the comparison shows the measurement methods performed and their results for the individual qualities of the hybrid components. In this way, the comparison shows how different qualities of the hybrid components change during the process due to the manufacturing methods used.

DOI: https://doi.org/10.48366/r187049

Data analysis with Juypter notebook



Properties	Investigation of the material combination 20MnCr5 and X45CrSi9-3 in the Tailored Forming of shafts with bearing seats *Air-cooled 2 Cladding Layer Component (x45CrSi9-3) - 2022*	Investigation of the material combination 20MnCr5 and X45CrSi9-3 in the Tailored Forming of shafts with bearing seats *Air-cooled 3 Cladding Layer Component (x45CrSi9-3) - 2022*	Investigation of the material combination 20MnCr5 and X45CrSi9-3 in the Tailored Forming of shafts with bearing seats *Water-cooled 2 Cladding Layer Component (x45CrSi9-3) - 2022*	Investigation of the material combination 20MnCr5 and X45CrSi9-3 in the Tailored Forming of shafts with bearing seats *Water-cooled 3 Cladding Layer Component (x45CrSi9-3) - 2022*	Cross-w welded rolling b material *1 Claddi (100Cr6)*
has material/material					
belongs to material group*	Steel	Steel	Steel	Steel	
has type*	20MnCr5 X45CrSi9-3	20MnCr5 X45CrSi9-3			

https://doi.org/10.48366/r187049

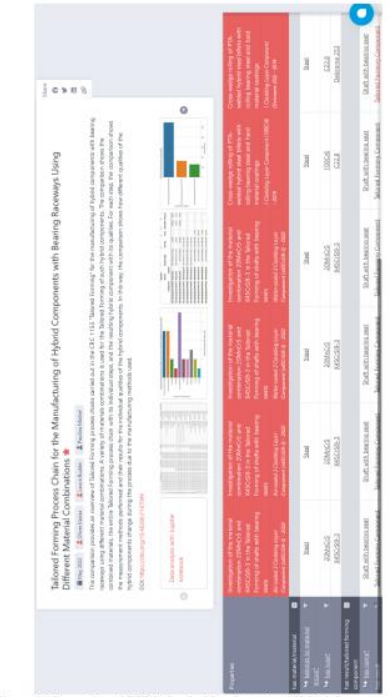


Figure 7: Comparison of TFPCs for the Manufacturing of Hybrid Components [38].

12

[11] Karras et al.: *Organizing Scientific Knowledge from Engineering Sciences Using the Open Research Knowledge Graph: The Tailored Forming Process Chain Use Case.* **Under Review** at Data Science Journal, 2024.

Example: ORKG Comparison of Literature: Software Features[12]

Comparison of Hyperparameter Optimization Tools

January 2023 | Oliver Karras | Difan Deng | Marius Lindauer

This comparison looks at the capabilities of various optimization tools for hyperparameter optimization, focusing on the following: Complex hyperparameter space, multi-objective, multi-fidelity, instances, command-line interface, and parallel computing. In addition, this comparison provides further information about the respective optimization tool, such as the locations of the code repository, README, software documentation and issue tracker, license and programming language.

DOI: https://doi.org/10.48366/r281265

Properties	BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization *BoTorch - 2020*	OpenBox: A Generalized Black-box Optimization Service *OpenBox - 2021*	BOHB: Robust and Efficient Hyperparameter Optimization at Scale *HpBandSter - 2018*	SMAC3: A Versatile Bayesian Optimization Package for Hyperparameter Optimization *SMAC3 - 2021*
result/software				
name*	BoTorch	OpenBox	HpBandSter	SMAC3
software features*	Multi-Fidelity Multi-Objective parallel computing	Complex Hyperparameter Space Multi-Objective parallel computing	Complex Hyperparameter Space Multi-Fidelity parallel computing	command-line interface Complex Hyperparameter Space Instances Multi-Fidelity Multi-Objective parallel computing
complex hyperparameter space*	✗	✓	✓	✓
multi-objective*	✓	✓	✗	✓
multi-fidelity*	✓	✗		
instance*	✗	✗		

https://doi.org/10.48366/r281265

Command-Line Interface

SMAC can not only be executed within a python file but also from the commandline. Consequently, not only algorithms in python can be optimized, but implementations in other languages as well.

Note

Command-line interface has been temporarily disabled in v2.0. Please fall back to v1.4 if you need it.

Comparison

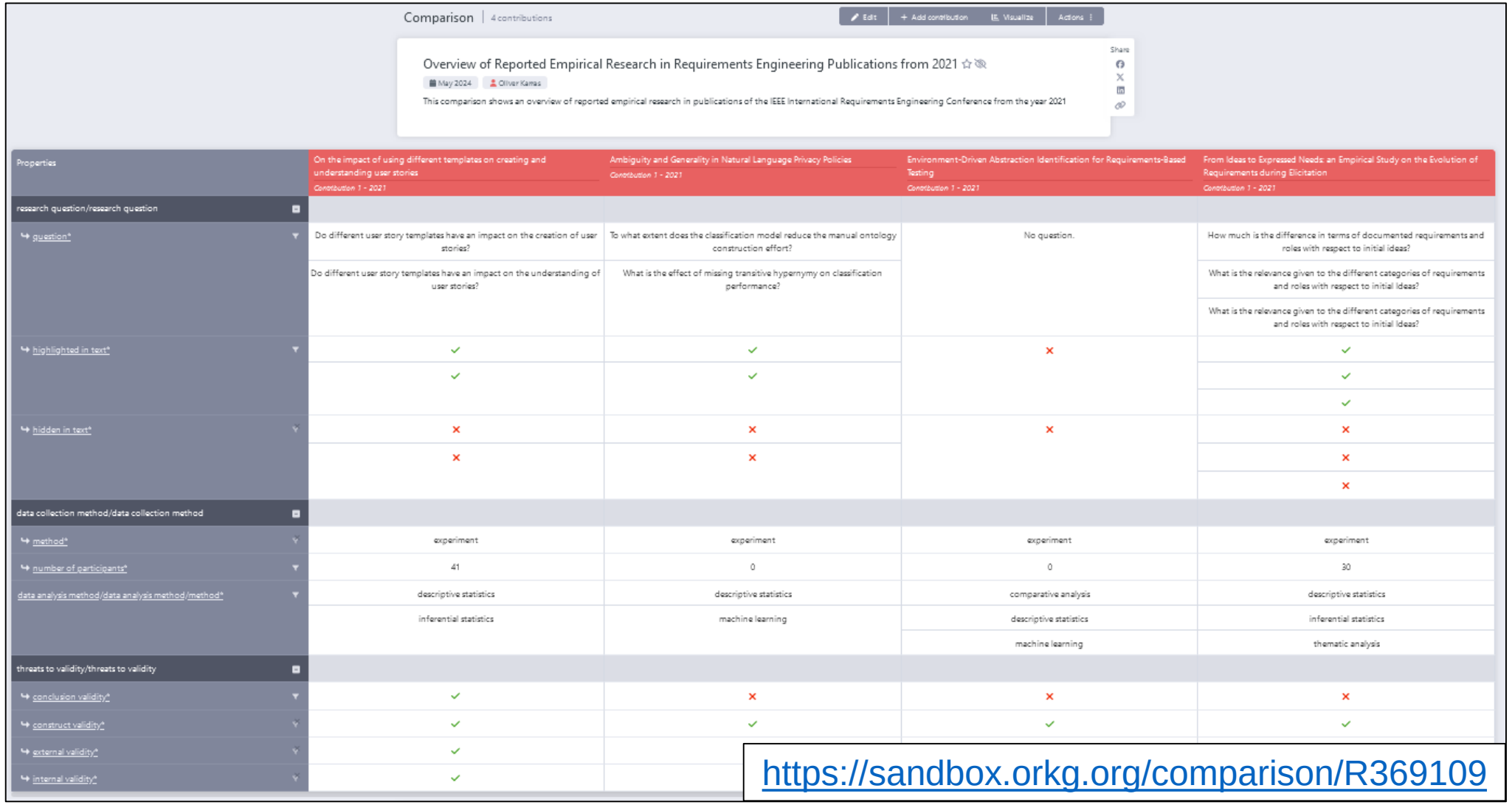
The following table provides an overview of SMAC's capabilities in comparison with other optimization tools.

Package	Complex Hyperparameter Space	Multi-Objective	Multi-Fidelity	Instances	Comma
HyperMapper	✓	✓	✗	✗	✗
Optuna	✓	✓	✓	✗	✓
Hyperopt	✓	✗	✗	✗	✓
BoTorch	✗	✓	✓	✗	✗
OpenBox	✓	✓	✗	✗	✗
HpBandSter	✓	✗	✓	✗	✗
SMAC	✓	✓	✓	✓	✓

Previous Installation | Next Getting Started

[12] Lindauer et al.: *SMAC3 Documentation*, URL: https://automl.github.io/SMAC3/v1.4.0/getting_started/package_overview.html#comparison, 2024.

Goal: ORKG Comparison on Reported Empirical Research



Comparison | 4 contributions

Edit | + Add contribution | Visualize | Actions

Overview of Reported Empirical Research in Requirements Engineering Publications from 2021

May 2024 | Oliver Karras

This comparison shows an overview of reported empirical research in publications of the IEEE International Requirements Engineering Conference from the year 2021

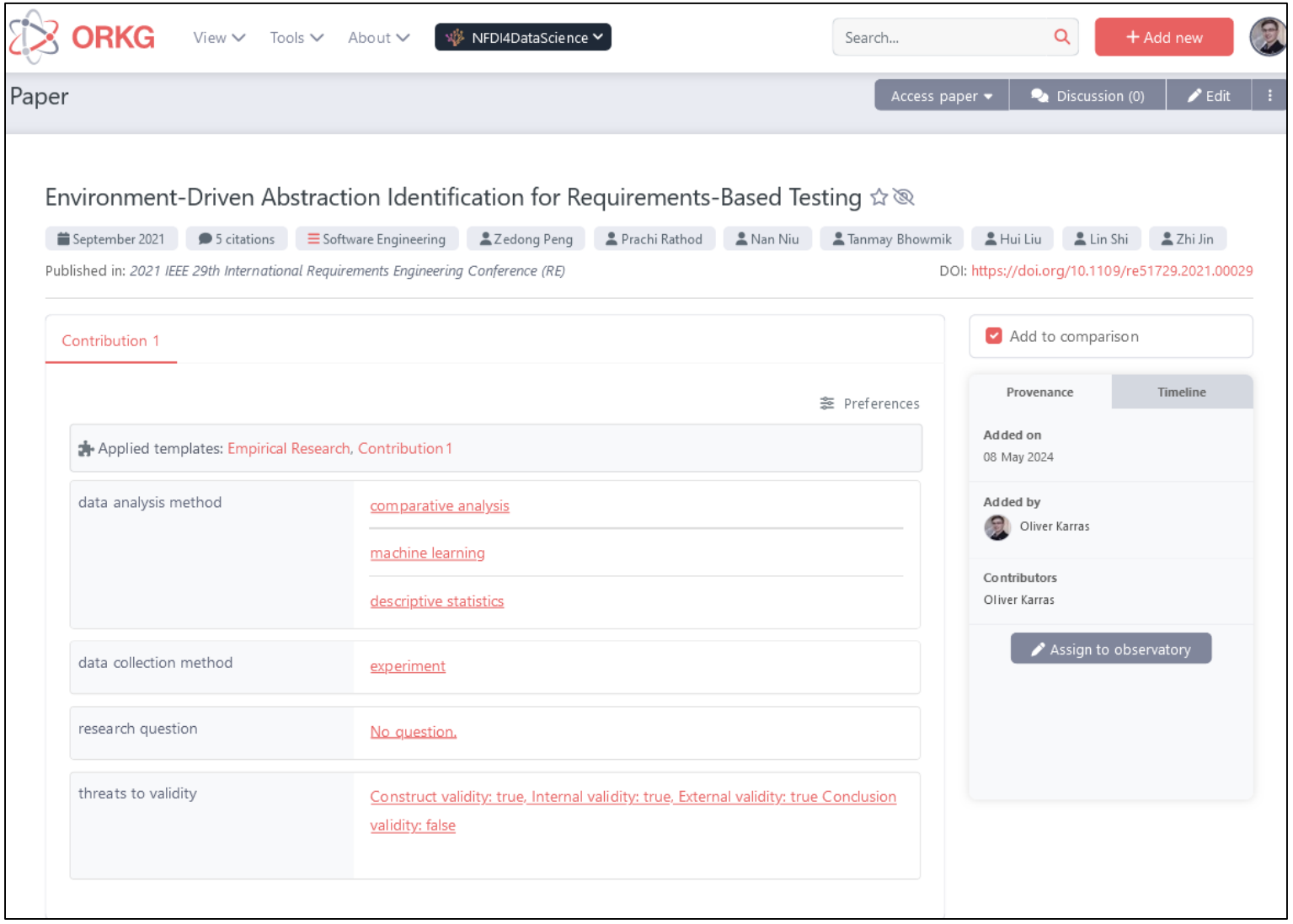
Share

Properties	On the impact of using different templates on creating and understanding user stories *Contribution 1 - 2021*	Ambiguity and Generality in Natural Language Privacy Policies *Contribution 1 - 2021*	Environment-Driven Abstraction Identification for Requirements-Based Testing *Contribution 1 - 2021*	From Ideas to Expressed Needs: an Empirical Study on the Evolution of Requirements during Elicitation *Contribution 1 - 2021*
research question/research question				
question*	Do different user story templates have an impact on the creation of user stories?	To what extent does the classification model reduce the manual ontology construction effort?	No question.	How much is the difference in terms of documented requirements and roles with respect to initial ideas?
	Do different user story templates have an impact on the understanding of user stories?	What is the effect of missing transitive hypernymy on classification performance?		What is the relevance given to the different categories of requirements and roles with respect to initial ideas?
				What is the relevance given to the different categories of requirements and roles with respect to initial ideas?
highlighted in text*	✓	✓	✗	✓
	✓	✓		✓
				✓
hidden in text*	✗	✗	✗	✗
	✗	✗		✗
				✗
data collection method/data collection method				
method*	experiment	experiment	experiment	experiment
number of participants*	41	0	0	30
data analysis method/data analysis method/method*	descriptive statistics	descriptive statistics	comparative analysis	descriptive statistics
	inferential statistics	machine learning	descriptive statistics	inferential statistics
			machine learning	thematic analysis
threats to validity/threats to validity				
conclusion validity*	✓	✗	✗	✗
construct validity*	✓	✓	✓	✓
external validity*	✓			
internal validity*	✓			

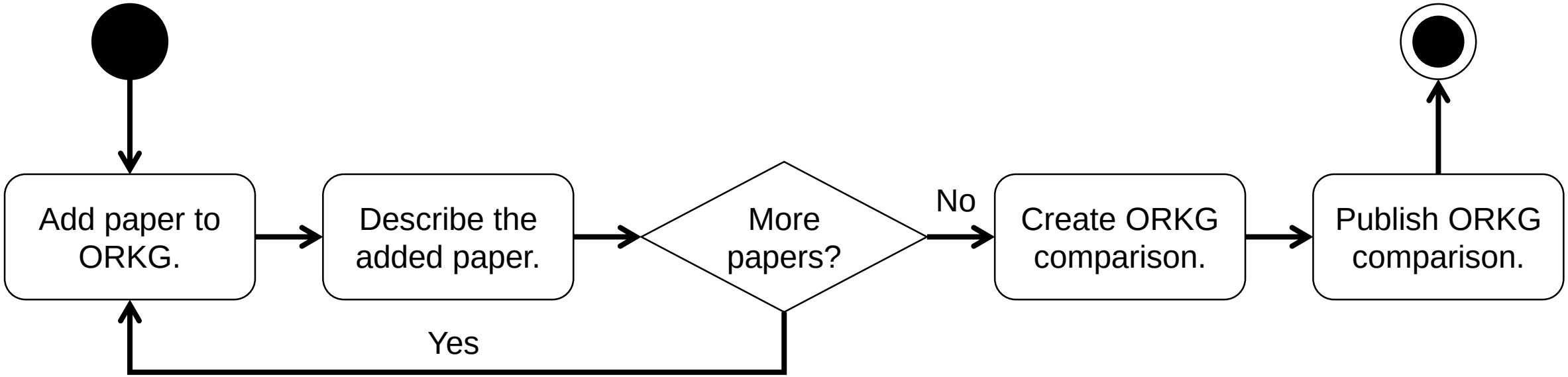
https://sandbox.orkg.org/comparison/R369109

Topic: Reported Empirical Research

- Empirical Research
  - *Research question*
    - Question
    - Hidden in text
    - Highlighted in text
  - *Data Collection Method*
    - Method
    - Number of participants
  - *Data Analysis Method*
    - Method
  - *Threats to Validity*
    - Construct Validity
    - Internal Validity
    - External Validity
    - Conclusion Validity



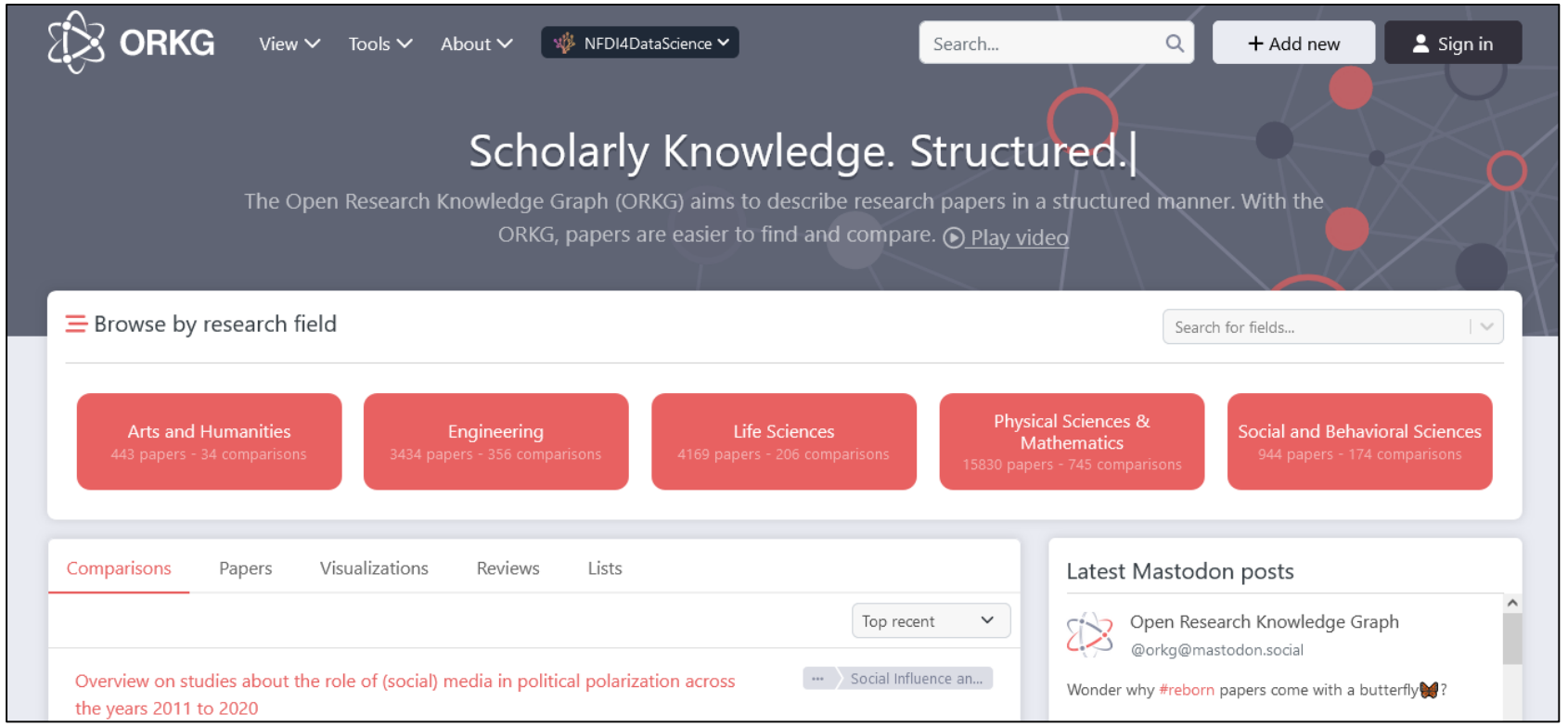
Overall Process for Creating an ORKG Comparison



Remark:
Instead of going through the loop, we will work **collaboratively** in the tutorial as conceived by ORKG. We will take all papers from all participants and compare them with each other in an ORKG comparison.

An ORKG comparison requires **at least two** publications.

1. Open ORKG Website



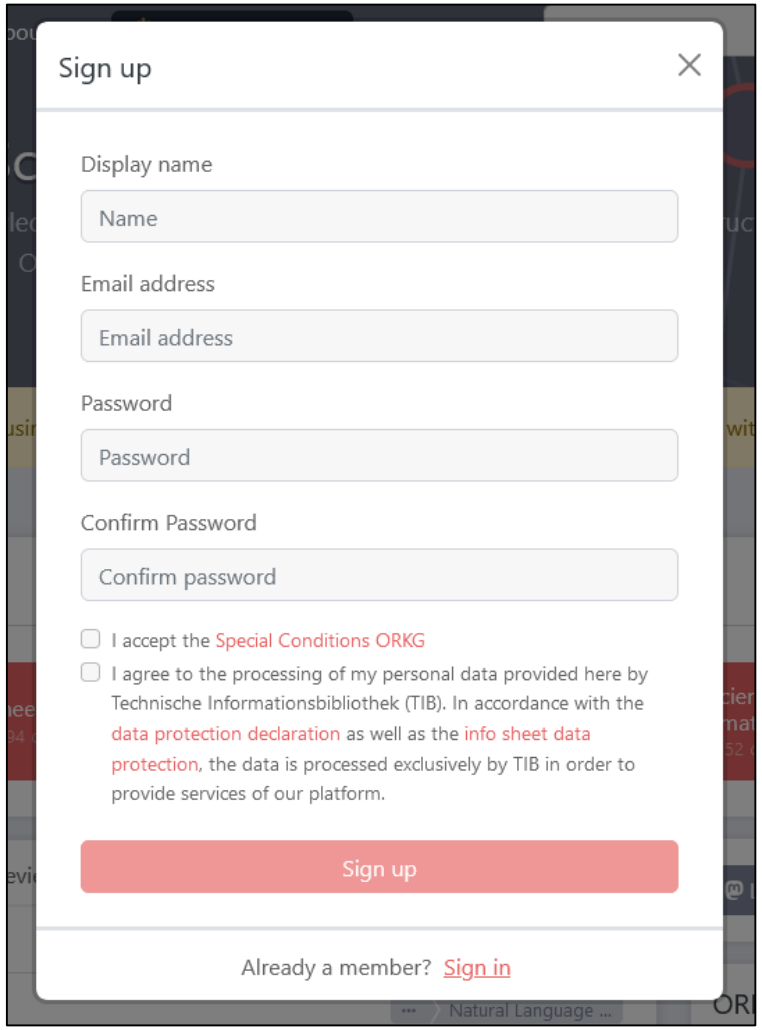
Remark:
For the tutorial, we use the **ORKG test environment** (https://sandbox.orkg.org/) to try everything without hesitation. You can find the official ORKG at https://orkg.org/.

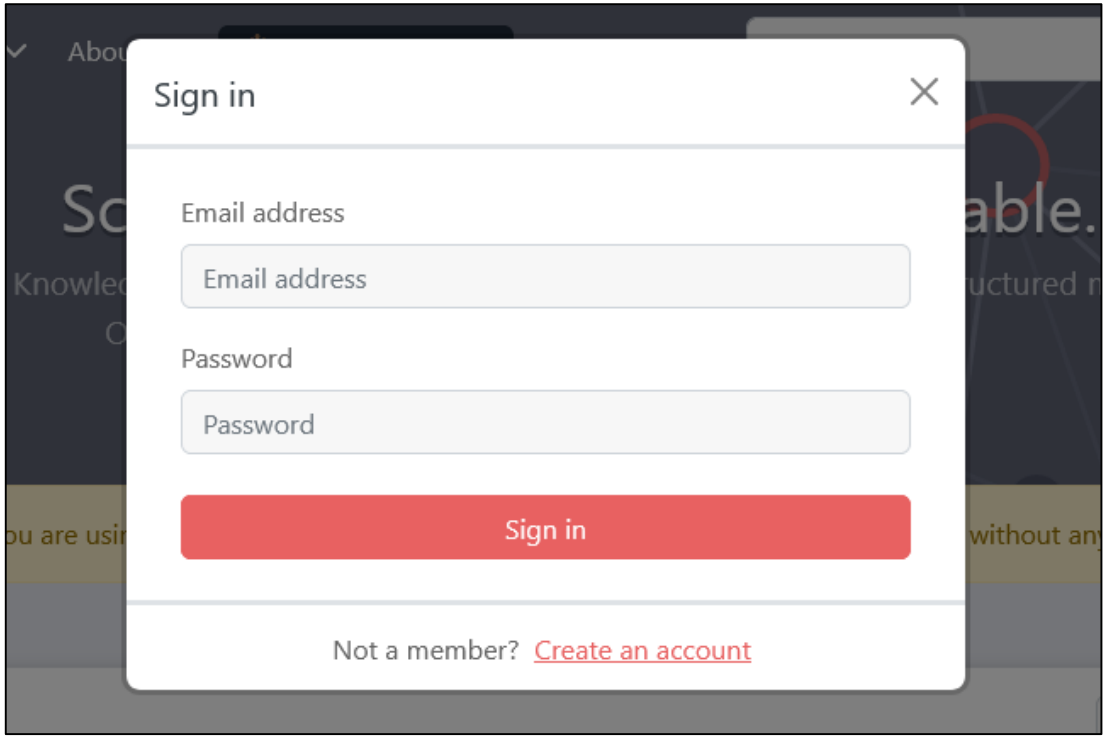
Get your exemplary paper: https://bit.ly/3UGFwhj

Use https://sandbox.orkg.org/!

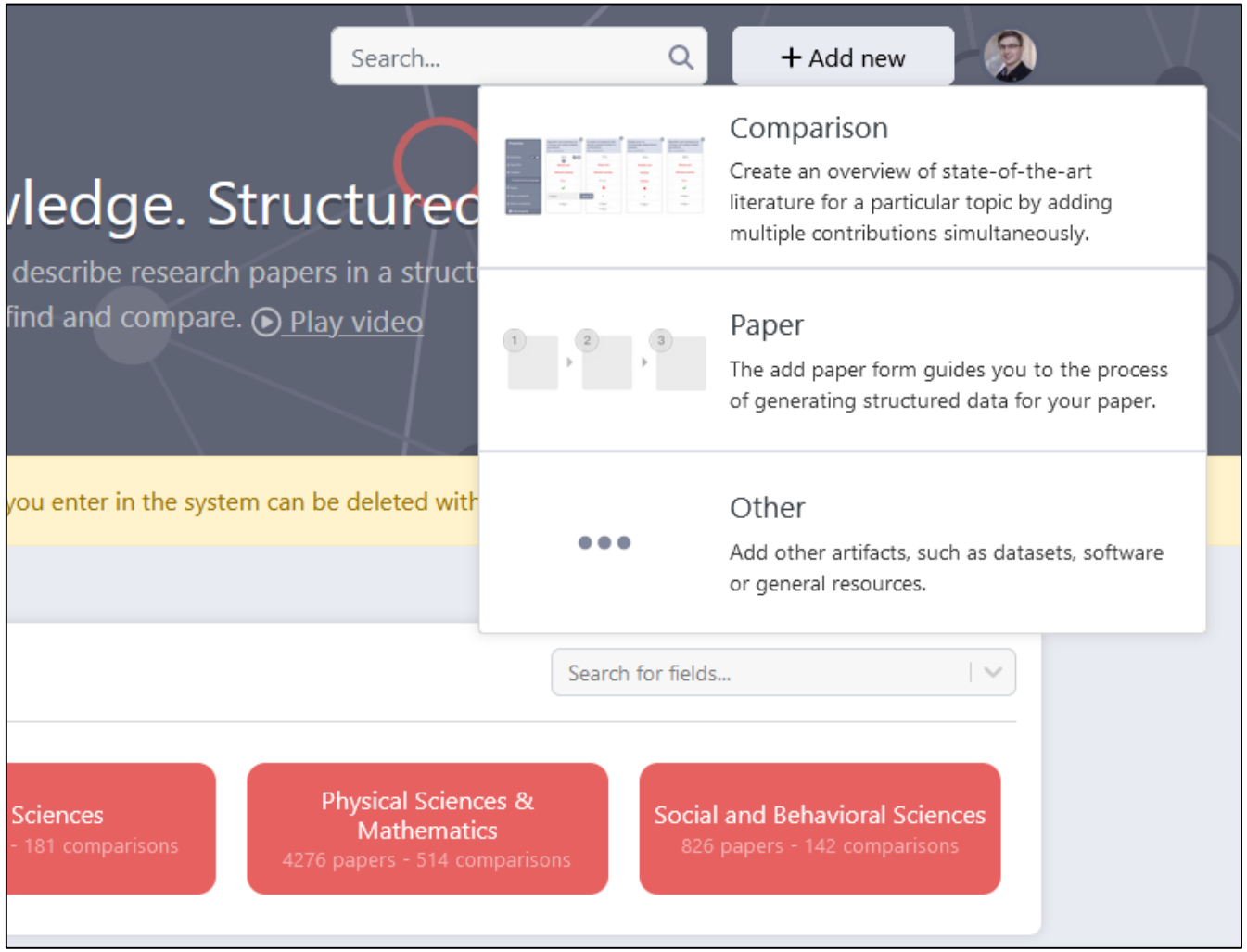
NOT https://orkg.org/!

2. Sign Up & Sign In





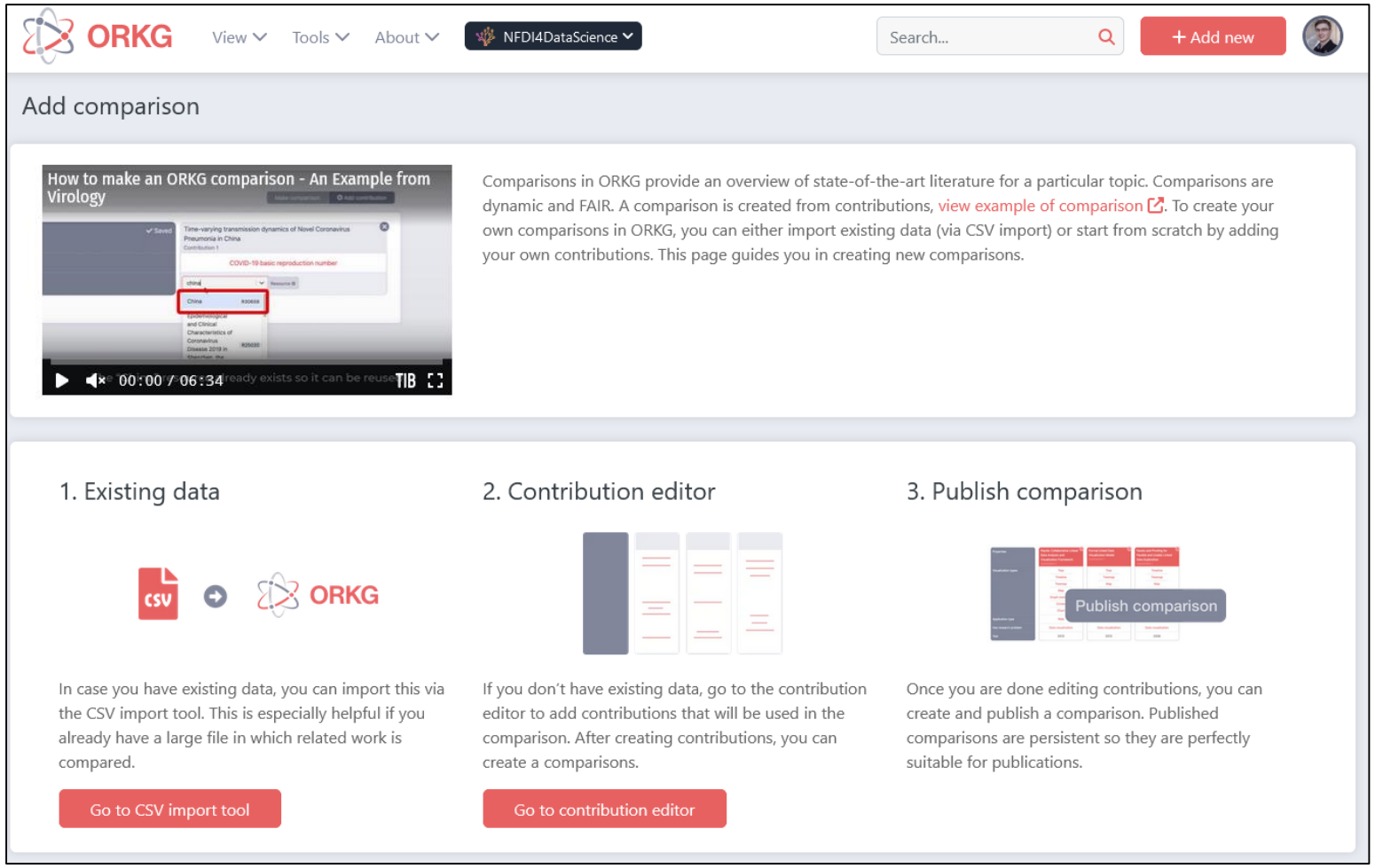
3. Add a New Comparison to the ORKG



1. Click on “+Add new”
2. Select “Comparison”

Remark:
We could also start by adding individual papers, but since we want to create a comparison at the end, it is easy to start immediately.

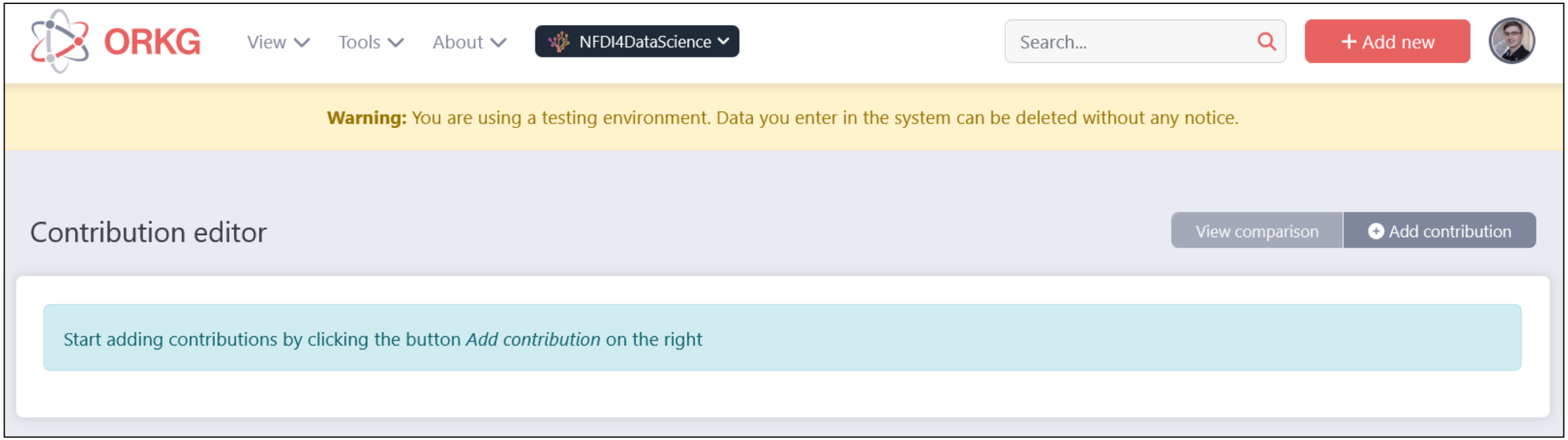
4. Go to the Contribution Editor

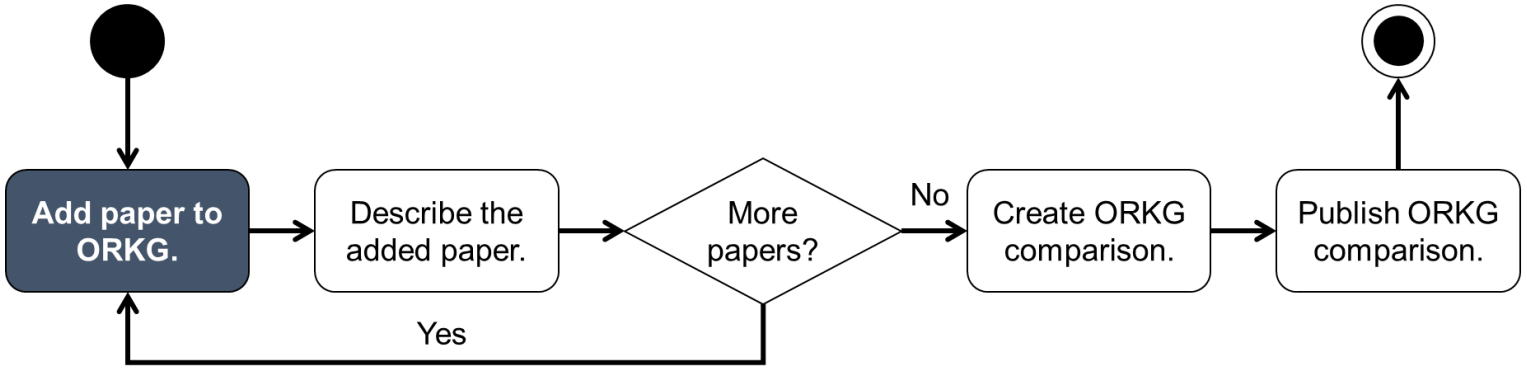


1. Click on “Go to contribution editor”

Remark:
When the data is already extracted, e.g., in a spreadsheet, we could use CSV import tool. However, the tool is a prototype and still requires a lot of manual work to map the data.

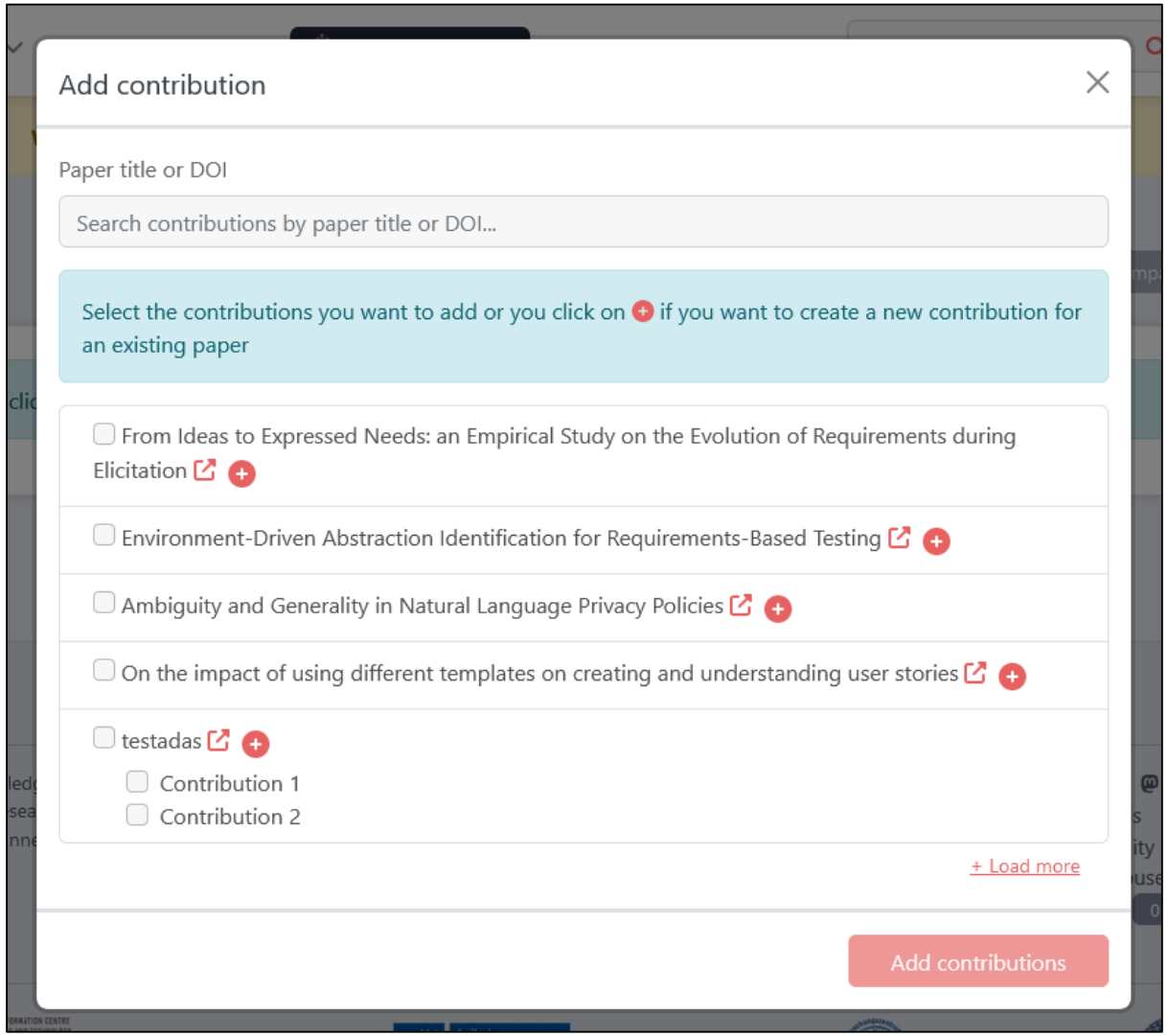
5. Add a Paper

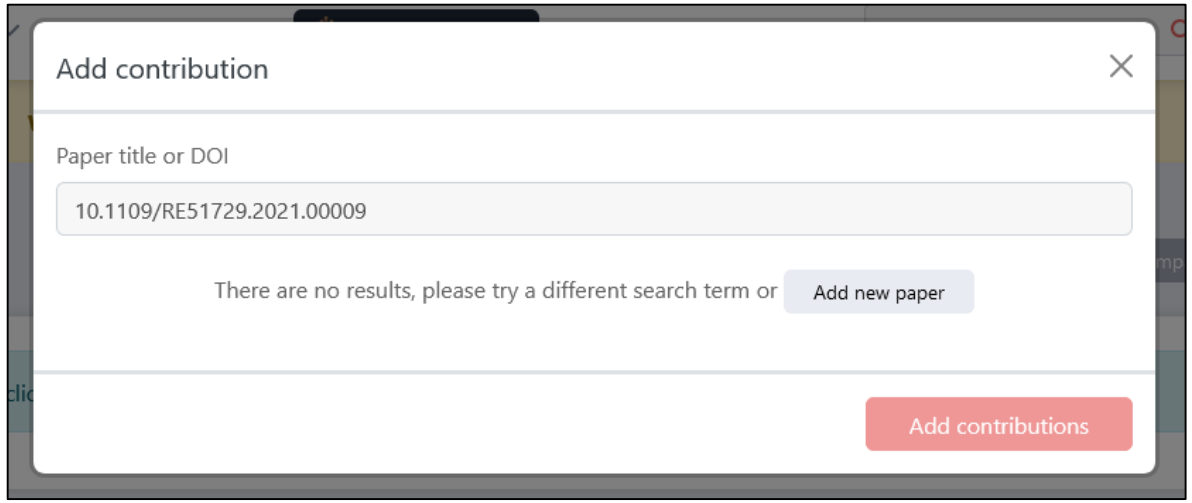




1. Click on “Add contribution”

5. Add a Paper: Enter DOI (or Paper Title)

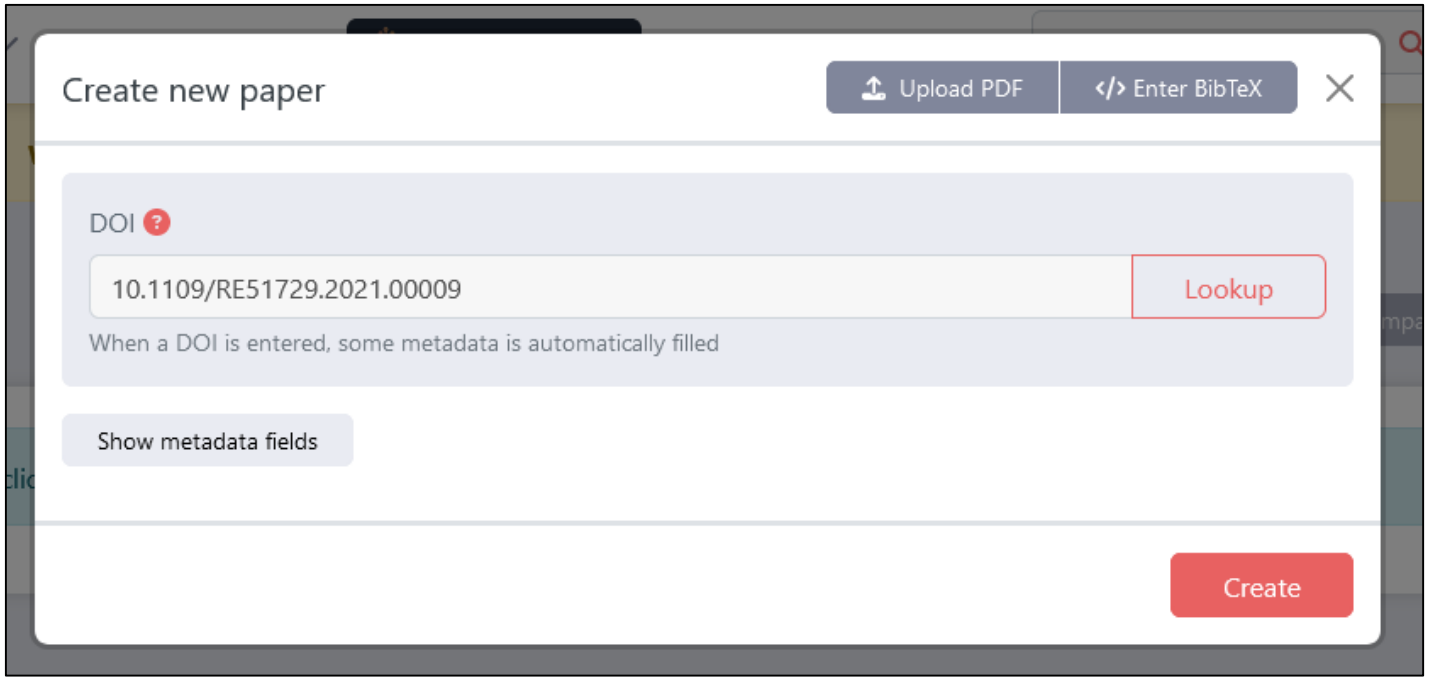




1. Click on “Add new paper”

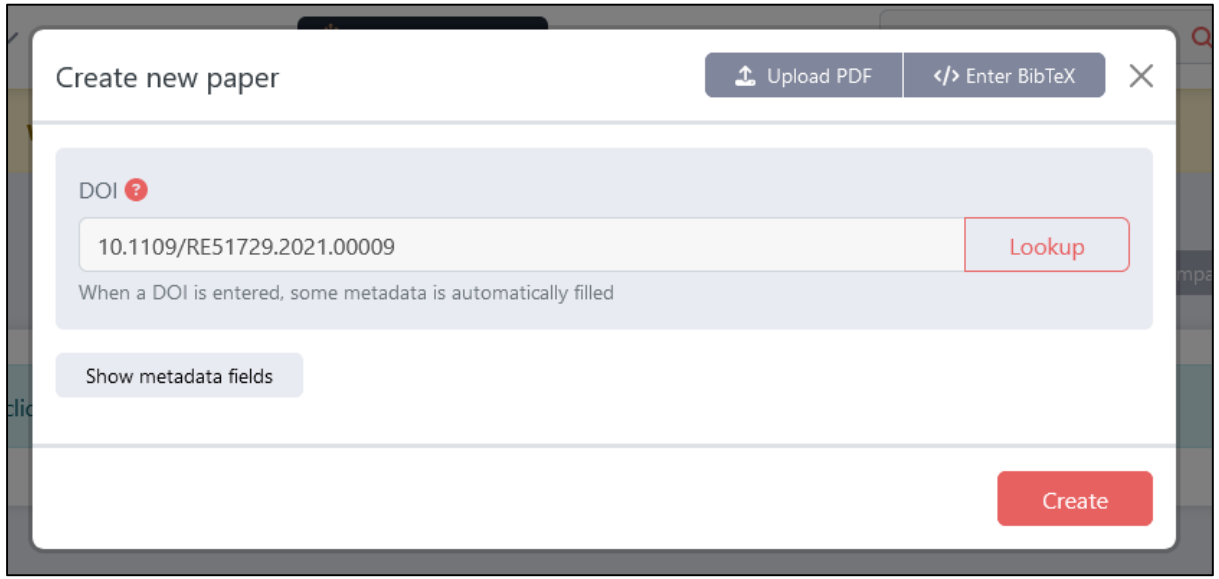
Remark:
If the paper is already in the ORKG, it is shown and we can either reuse an existing contribution or create a new one.

5. Add a Paper: Lookup Paper by DOI

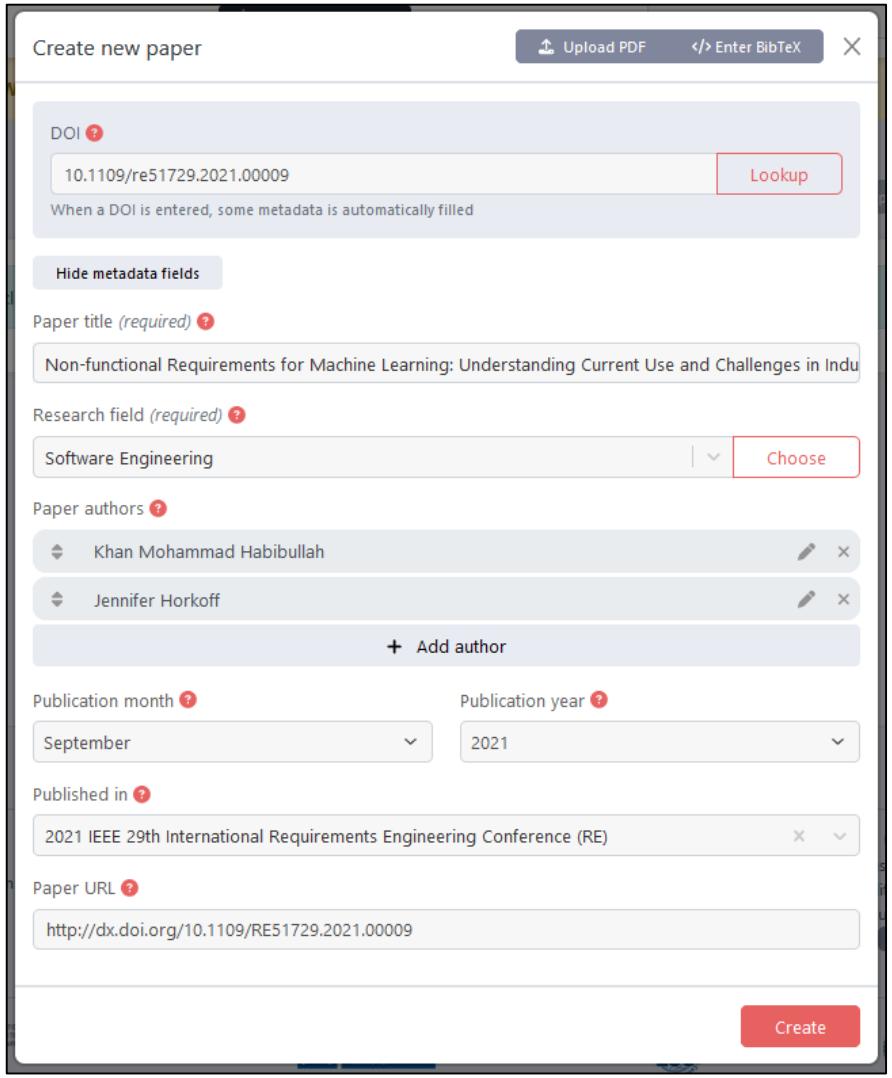


Remark:
ORKG attempts to retrieve the metadata automatically. If the retrieval is not successful, you can also manually enter or correct the metadata.

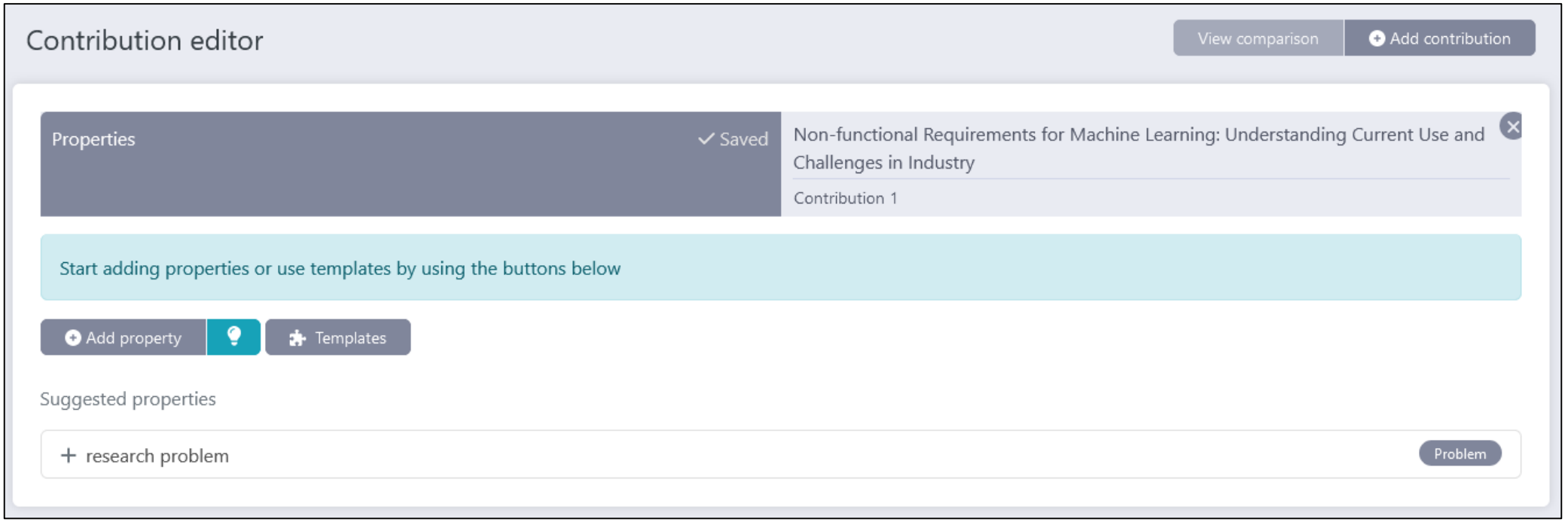
5. Add a Paper: Lookup Paper by DOI

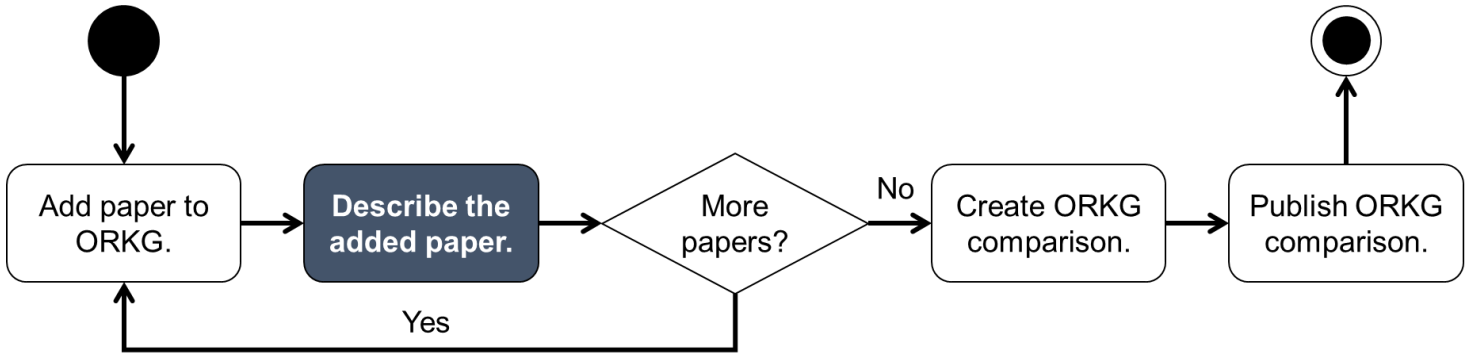


1. Click on “Lookup”
2. Check metadata fetched
3. Add a “Research field”. We use “Software Engineering”.
4. Click on “Create”



6. Describe a Paper



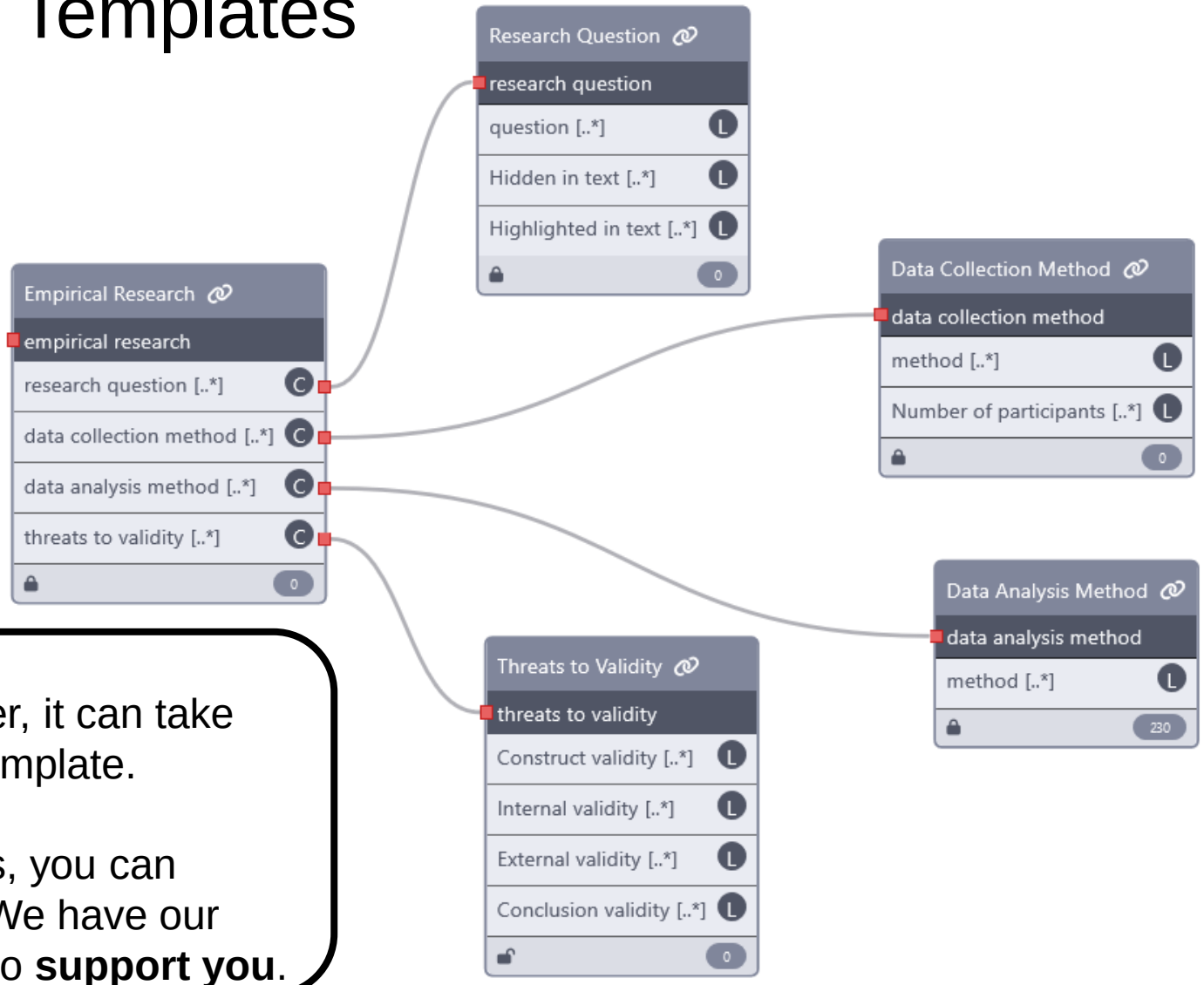


Remark:
Instead of adding all properties individually, we use an ORKG template.

6. Describe a Paper: ORKG Templates

An **ORKG template** specifies the **structure** and **terminology** used to describe a **publication** in the ORKG. Its use **ensures FAIR data** that is **consistent** and **comparable** across publications.

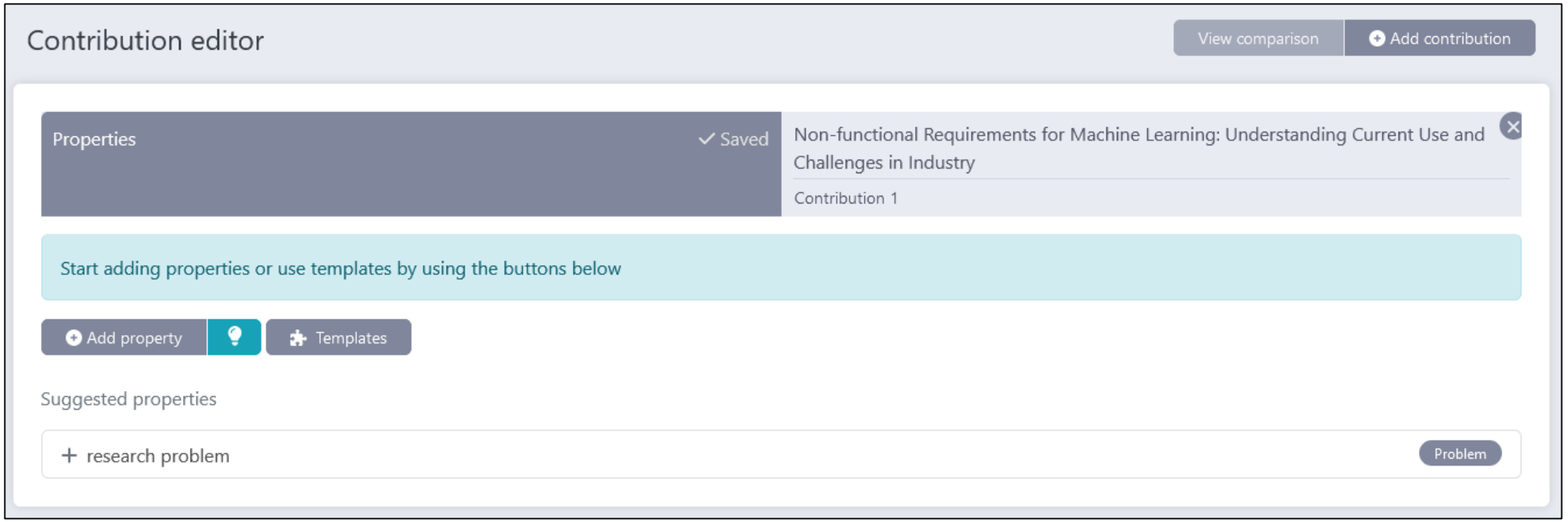
https://sandbox.orkg.org/template/R369028



Remark:
Every user can create ORKG templates. However, it can take some time and practice to build a good ORKG template.

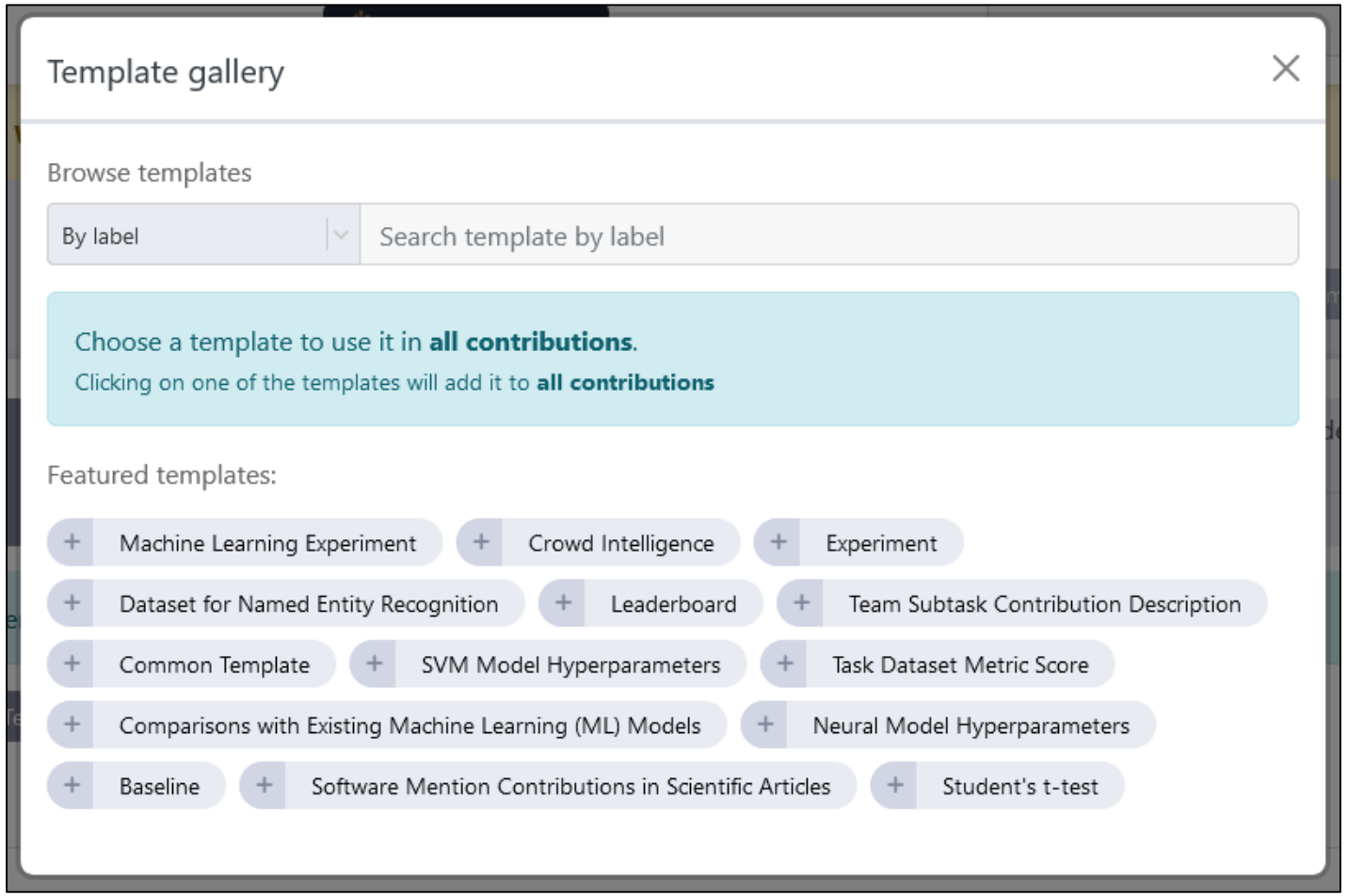
If you want to use the feature and have problems, you can always contact the ORKG team, especially me. We have our own **Curation and Community Building team** to **support you**.

6. Describe a Paper: Add Template “Empirical Research”



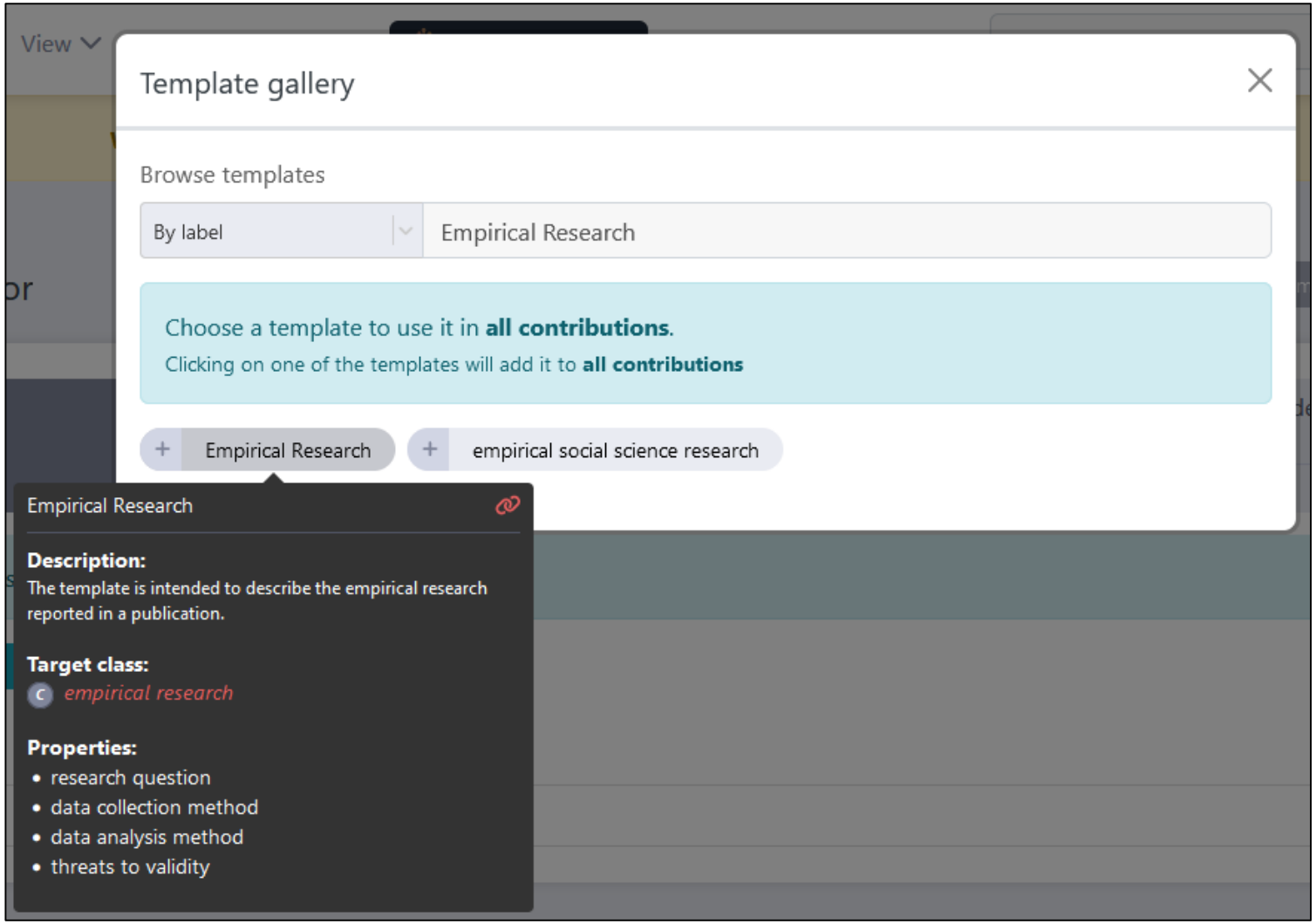
1. Click on “Templates”

6. Describe a Paper: Add Template “Empirical Research”



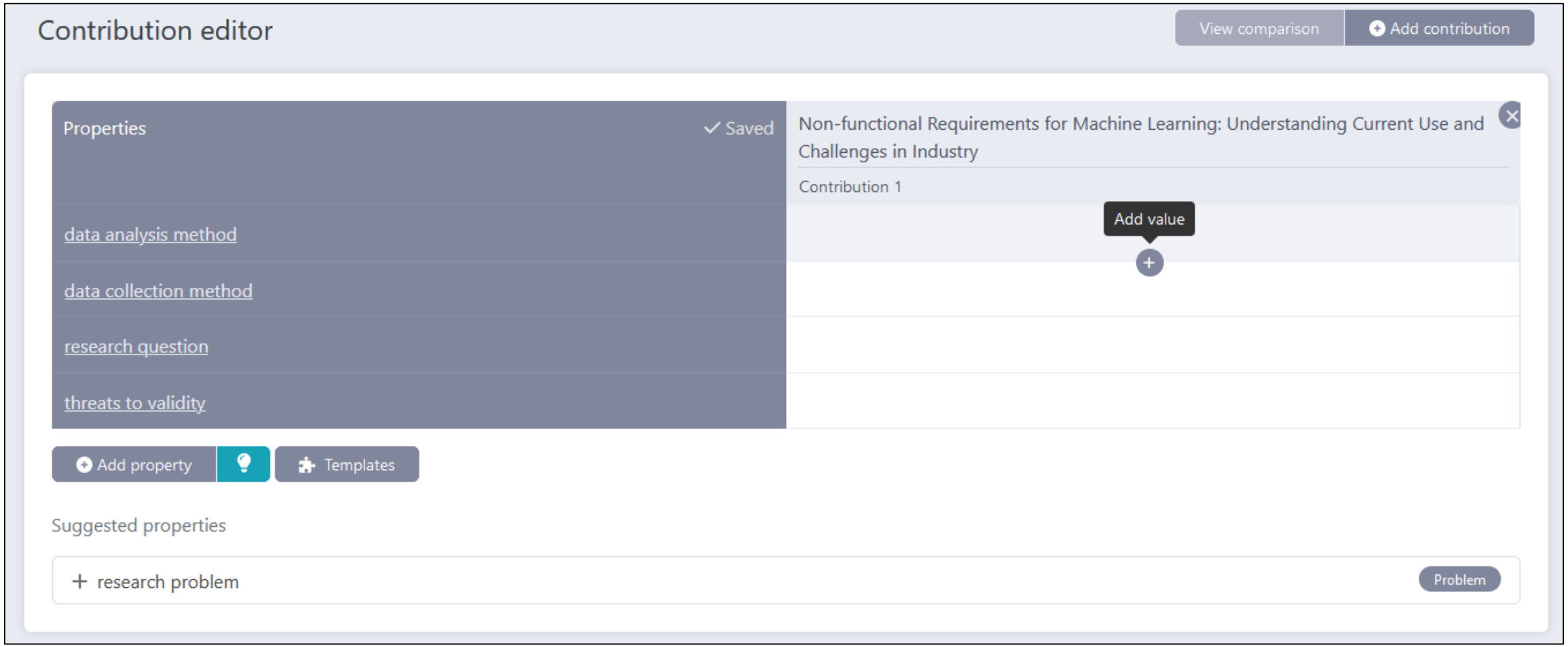
1. Search for “Empirical Research”

6. Describe a Paper: Add Template “Empirical Research”



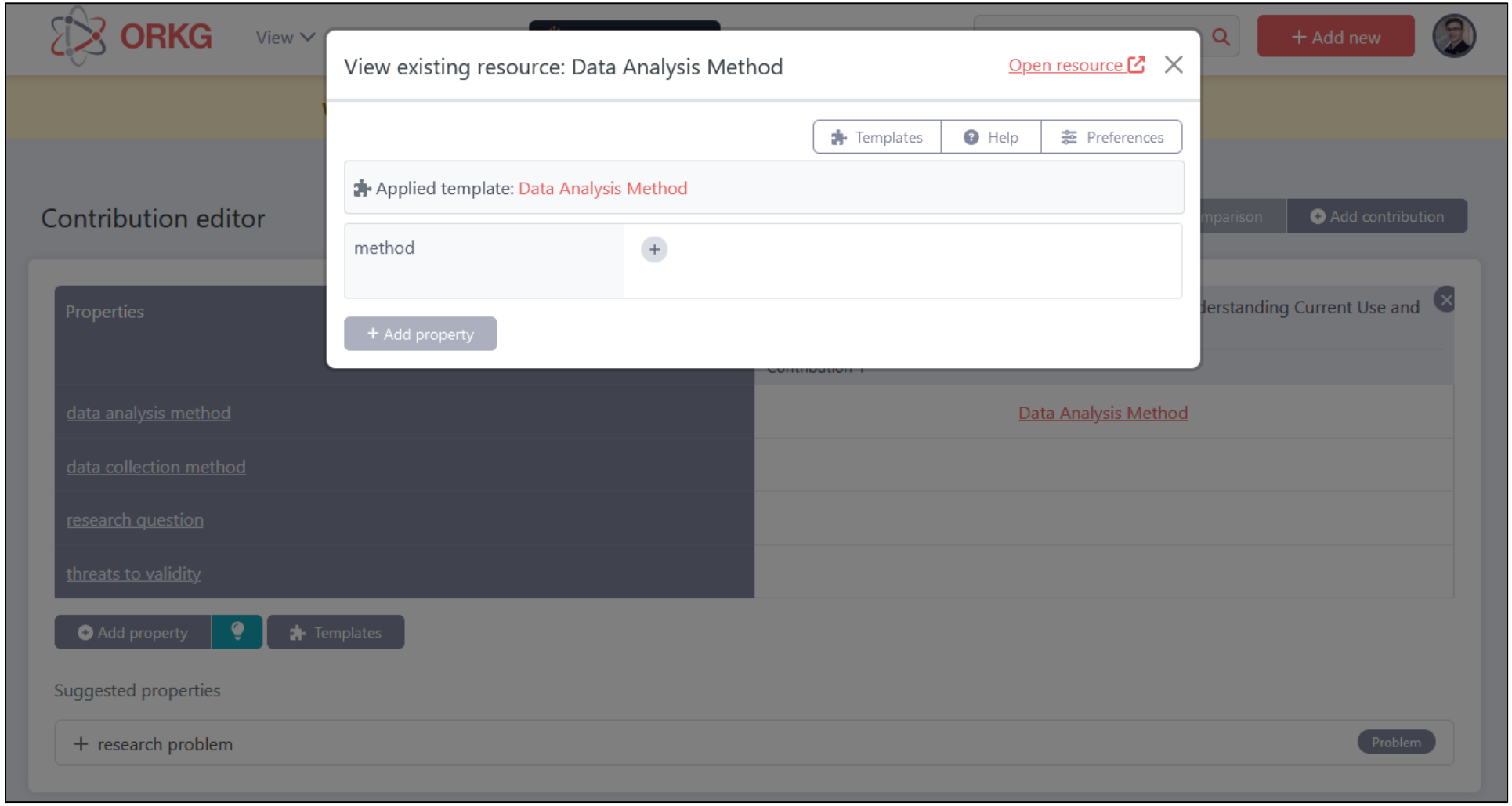
1. Select “Empirical Research”

6. Describe a Paper: Fill out the Table



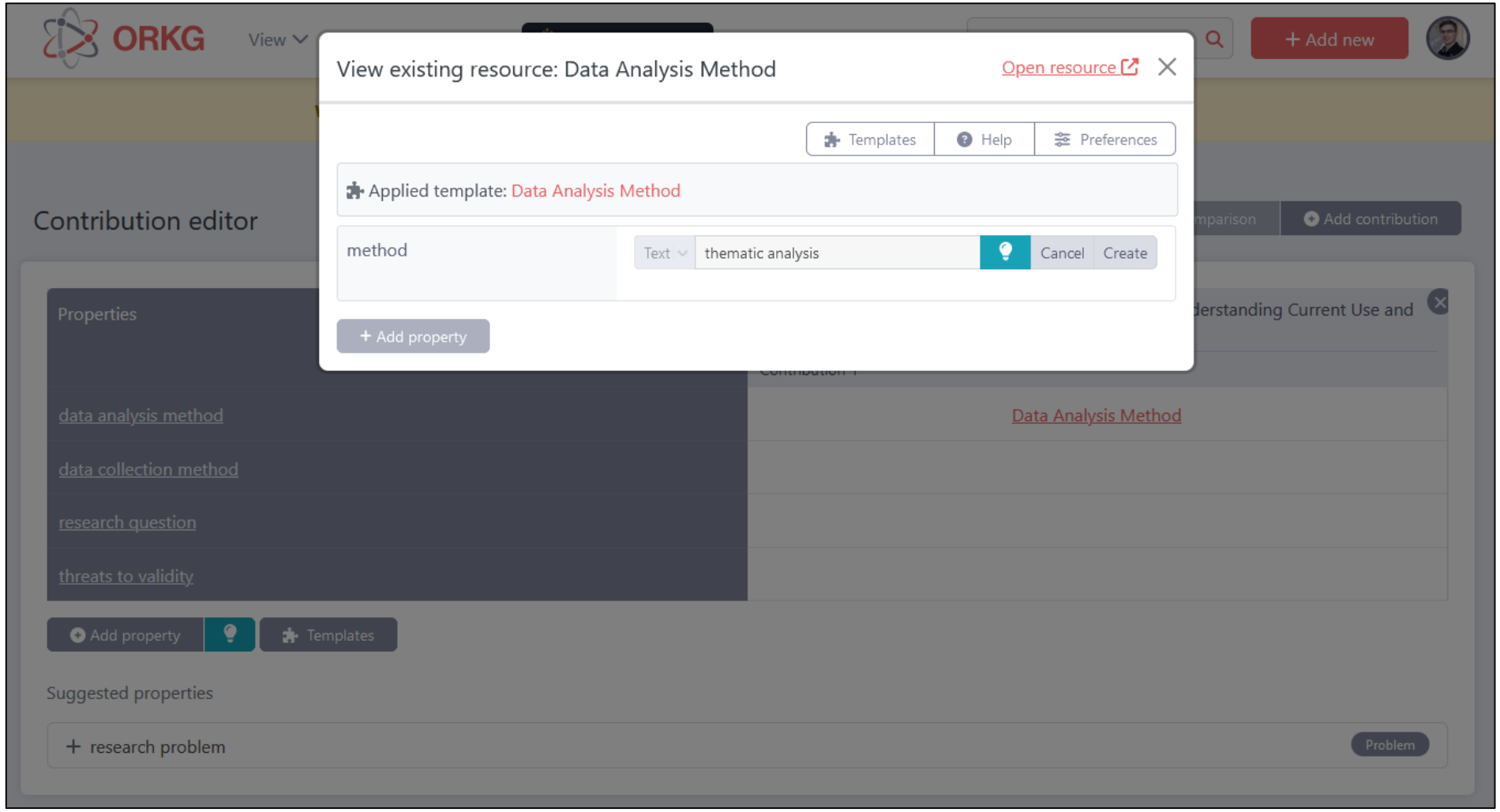
1. Click “Add value” for adding the data analysis method

6. Describe a Paper: Enter the Data Analysis Method(s)



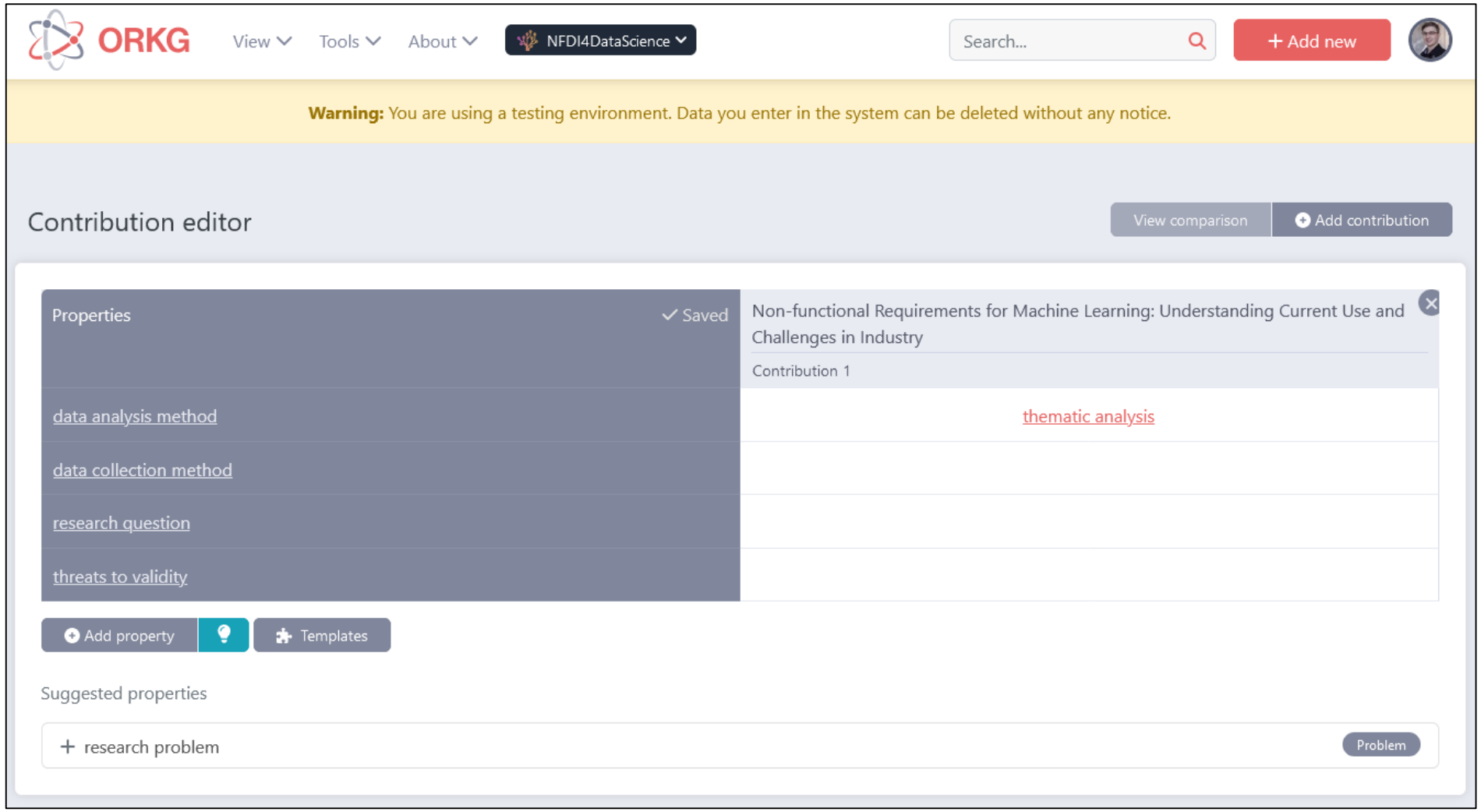
1. Click the ⊕
2. Enter method

6. Describe a Paper: Enter the Data Analysis Method(s)



1. Click “Create”
2. Close dialog

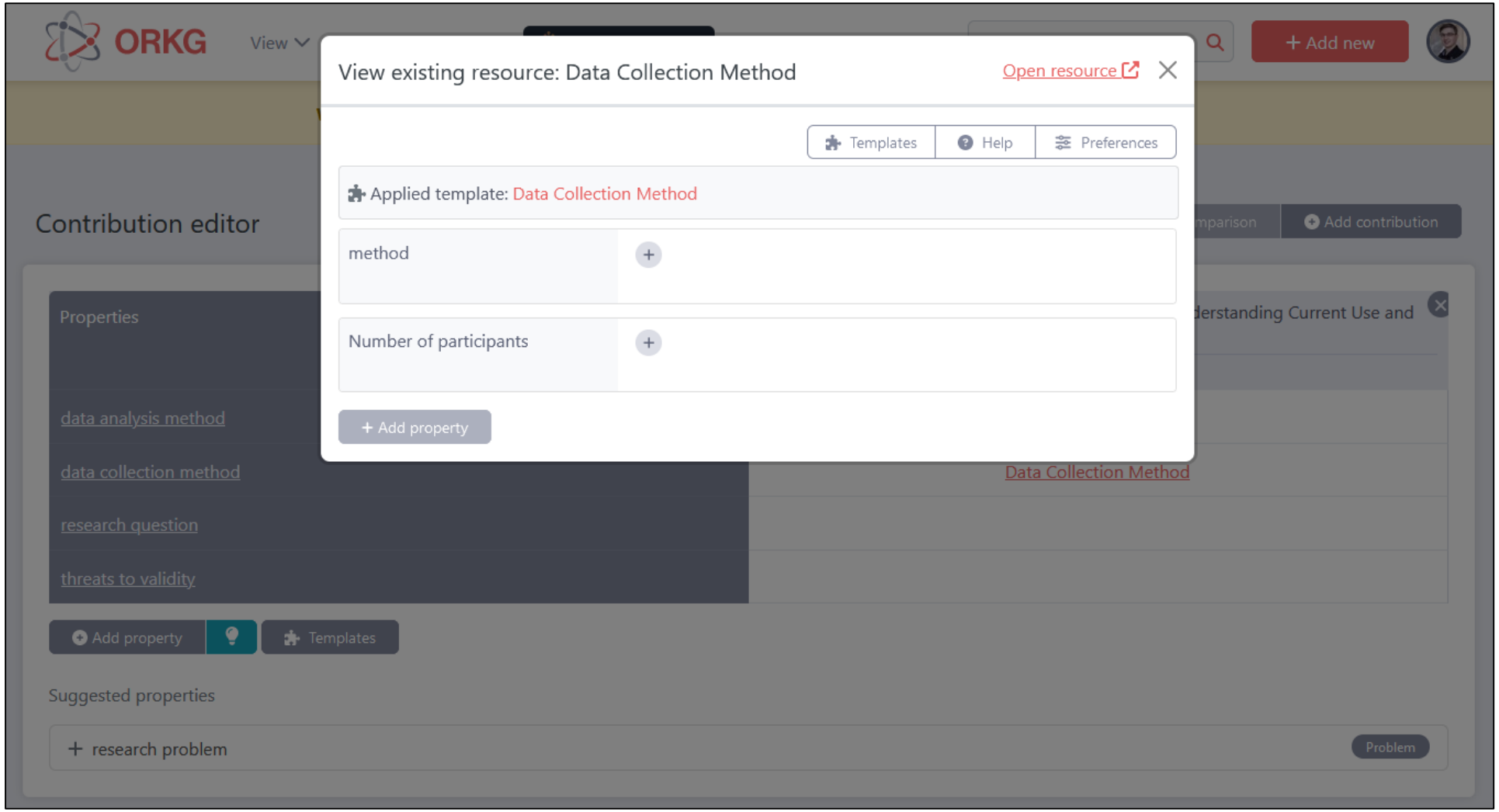
6. Describe a Paper: Enter the Data Analysis Method(s)



If you have more analysis methods, **repeat** the process.

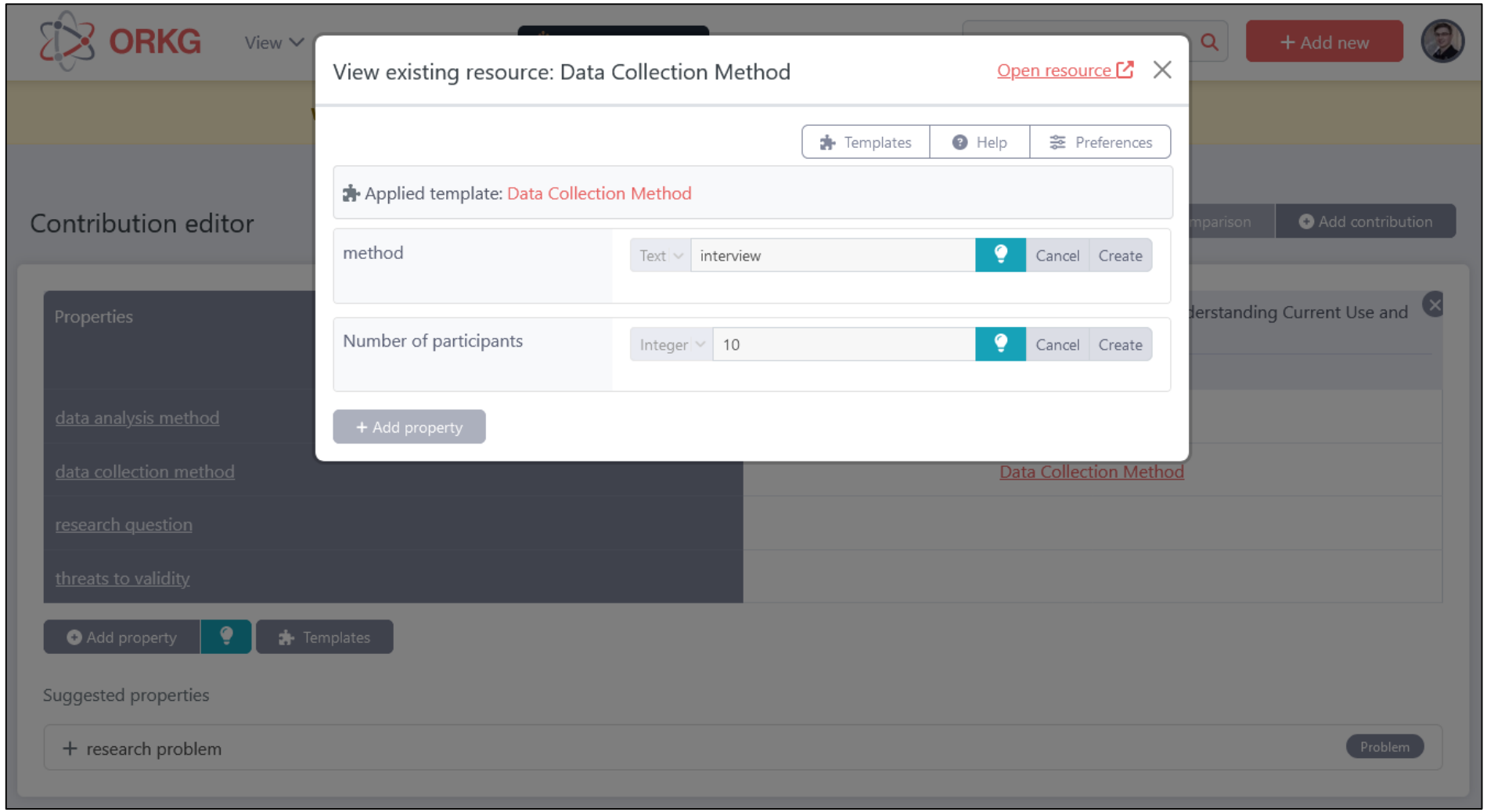
Otherwise, continue with the next property.

6. Describe a Paper: Enter the Data Collection Method(s)



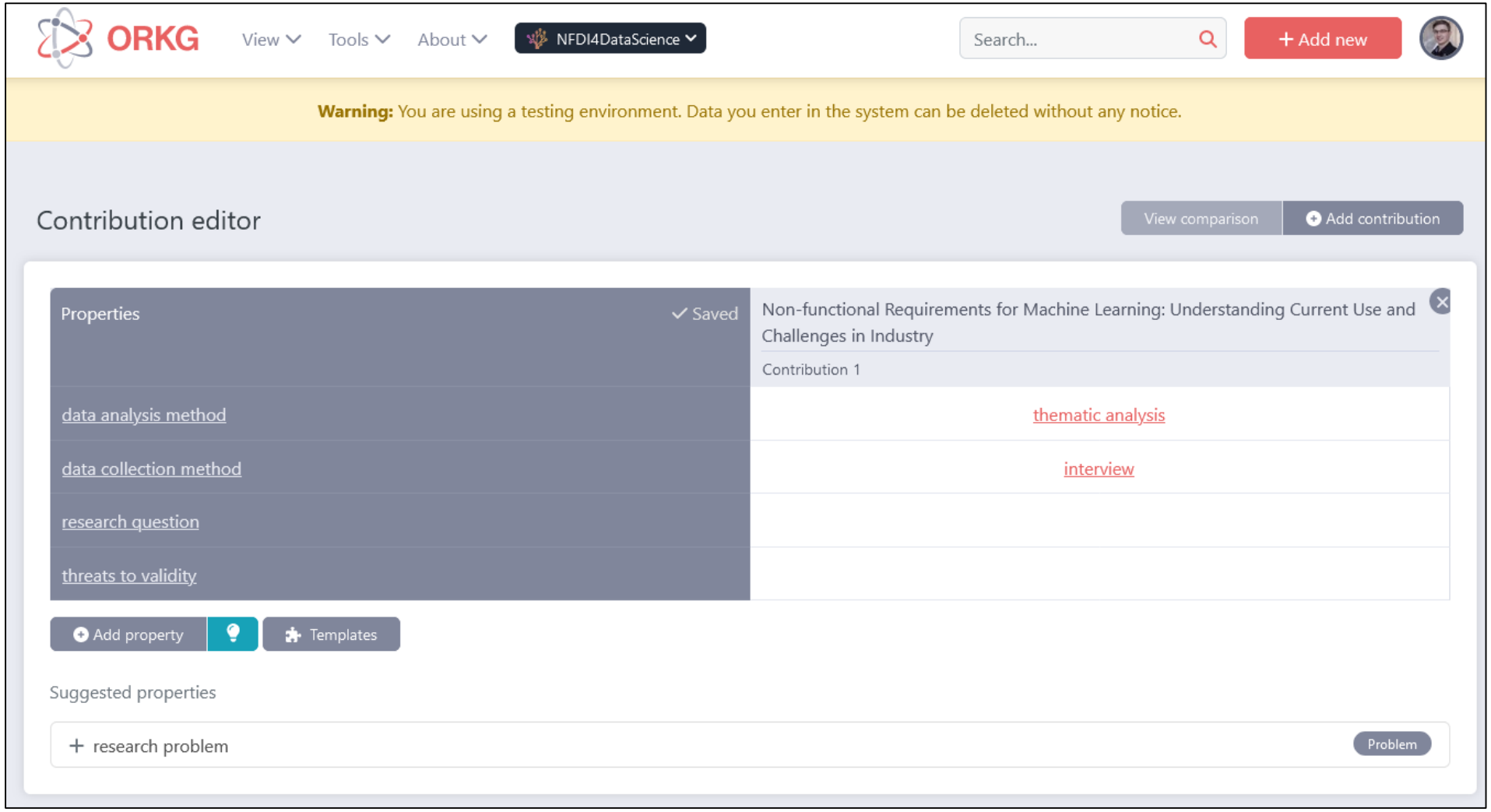
1. Click the ⊕
2. Enter method
3. Click the ⊕
4. Enter number of participants

6. Describe a Paper: Enter the Data Collection Method(s)



1. Click “Create”
2. Click “Create”
3. Close dialog

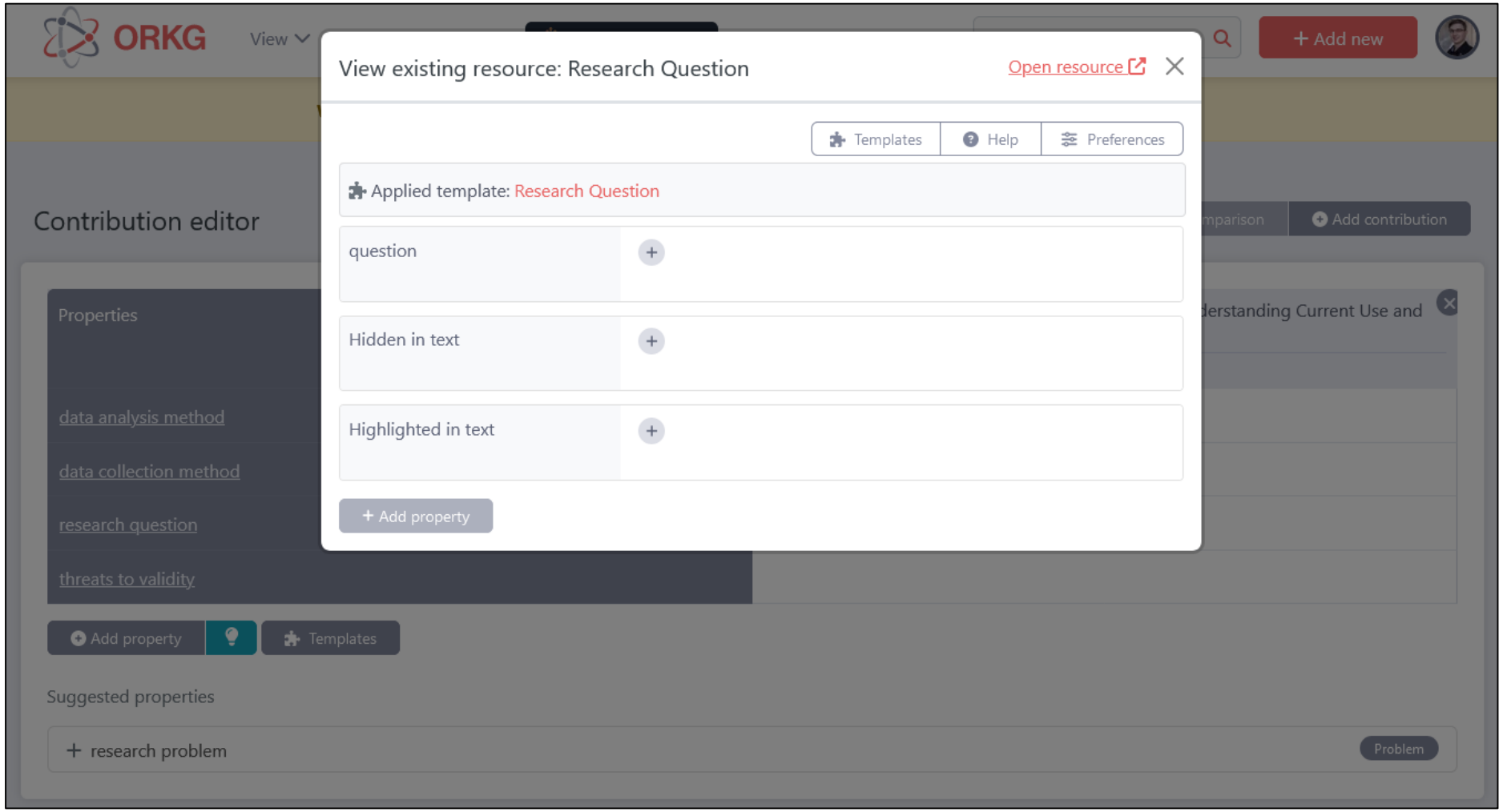
6. Describe a Paper: Enter the Data Collection Method(s)



If you have more collection methods, **repeat** the process.

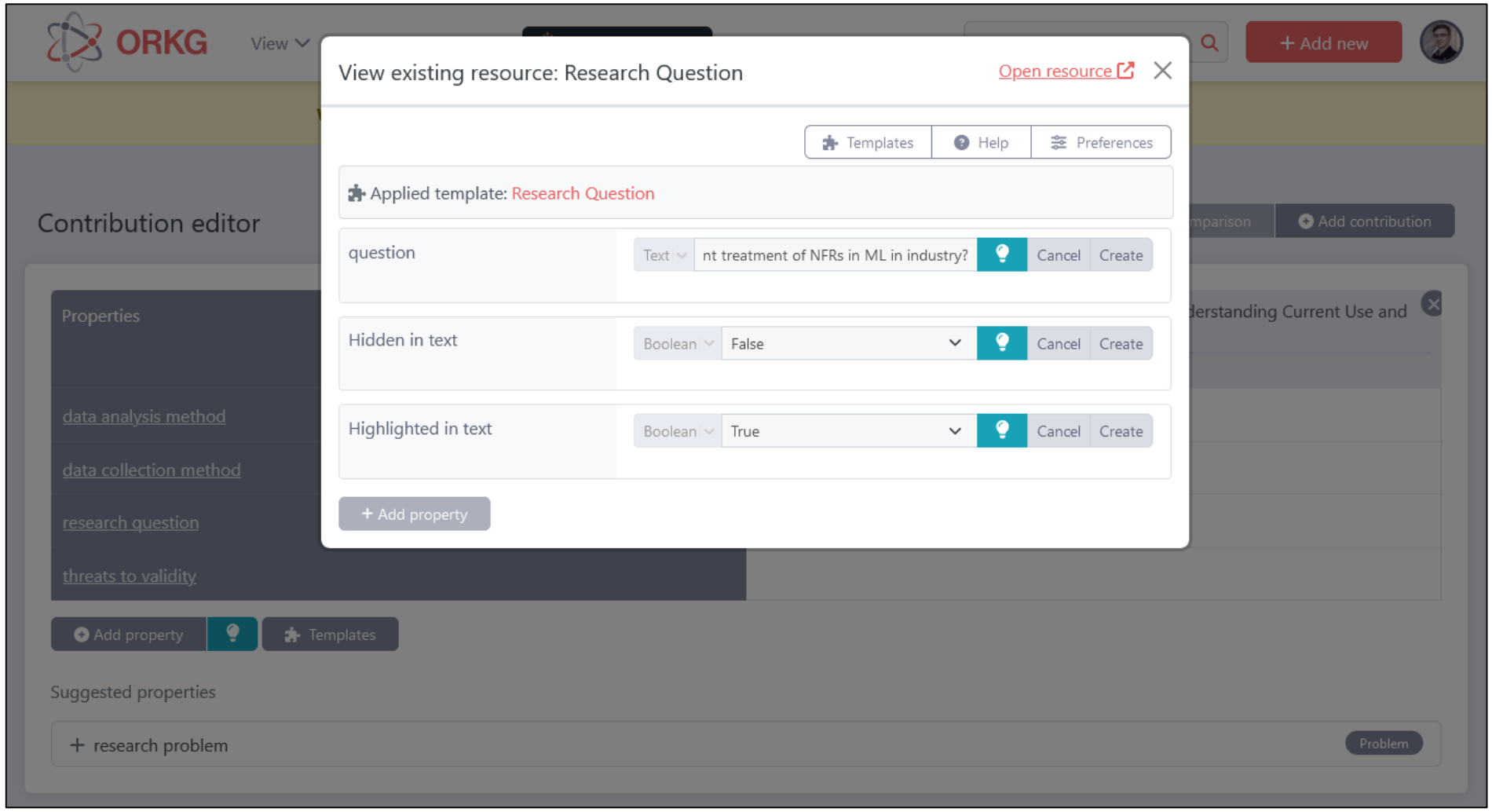
Otherwise, continue with the next property.

6. Describe a Paper: Enter the Research Question(s)



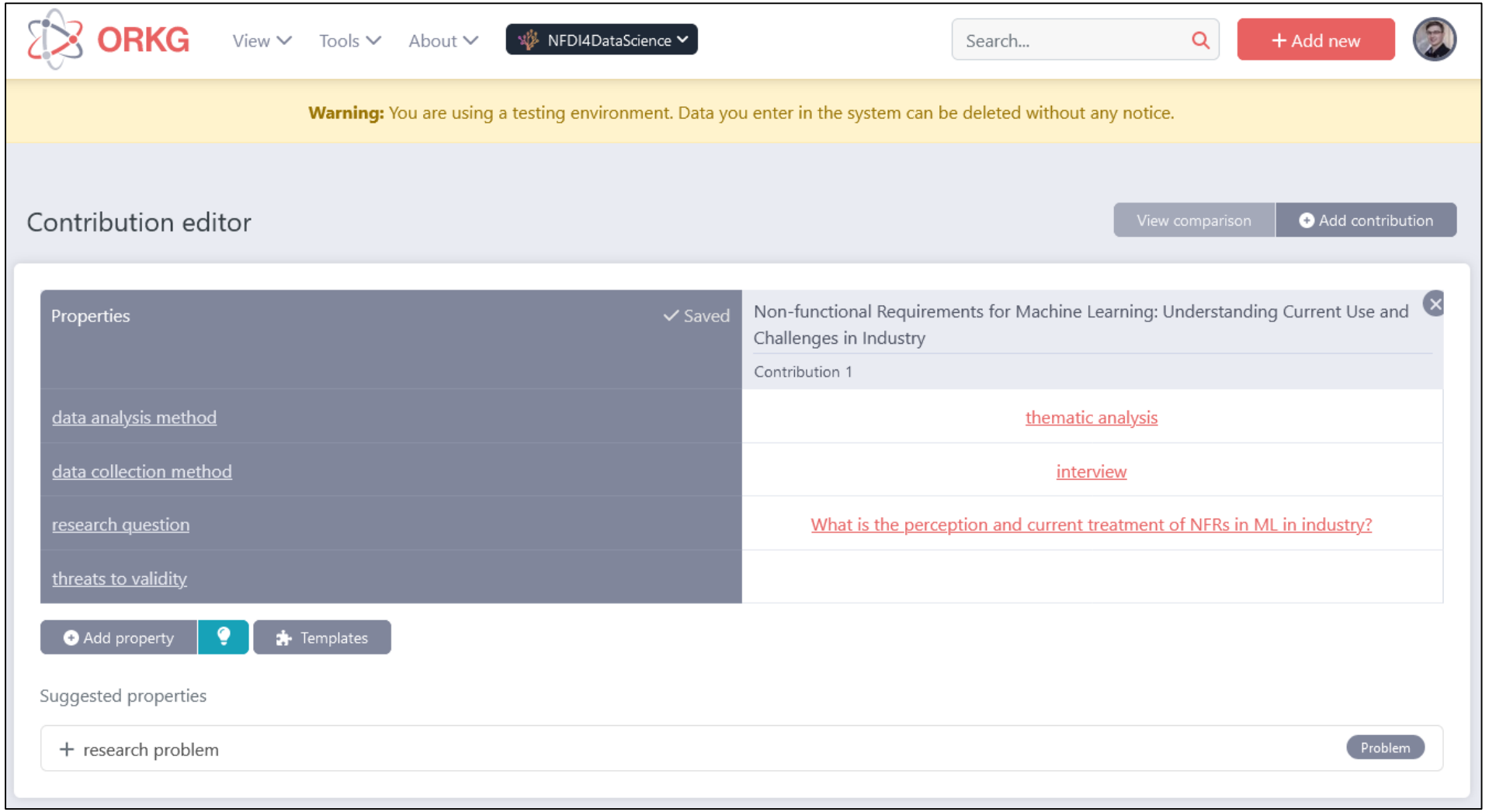
1. Click the ⊕
2. Enter question
3. Click the ⊕
4. Enter, if the question is hidden in text
5. Click the ⊕
6. Enter, if the question is highlighted in text

6. Describe a Paper: Enter the Research Question(s)



1. Click “Create”
2. Click “Create”
3. Click “Create”
4. Close dialog

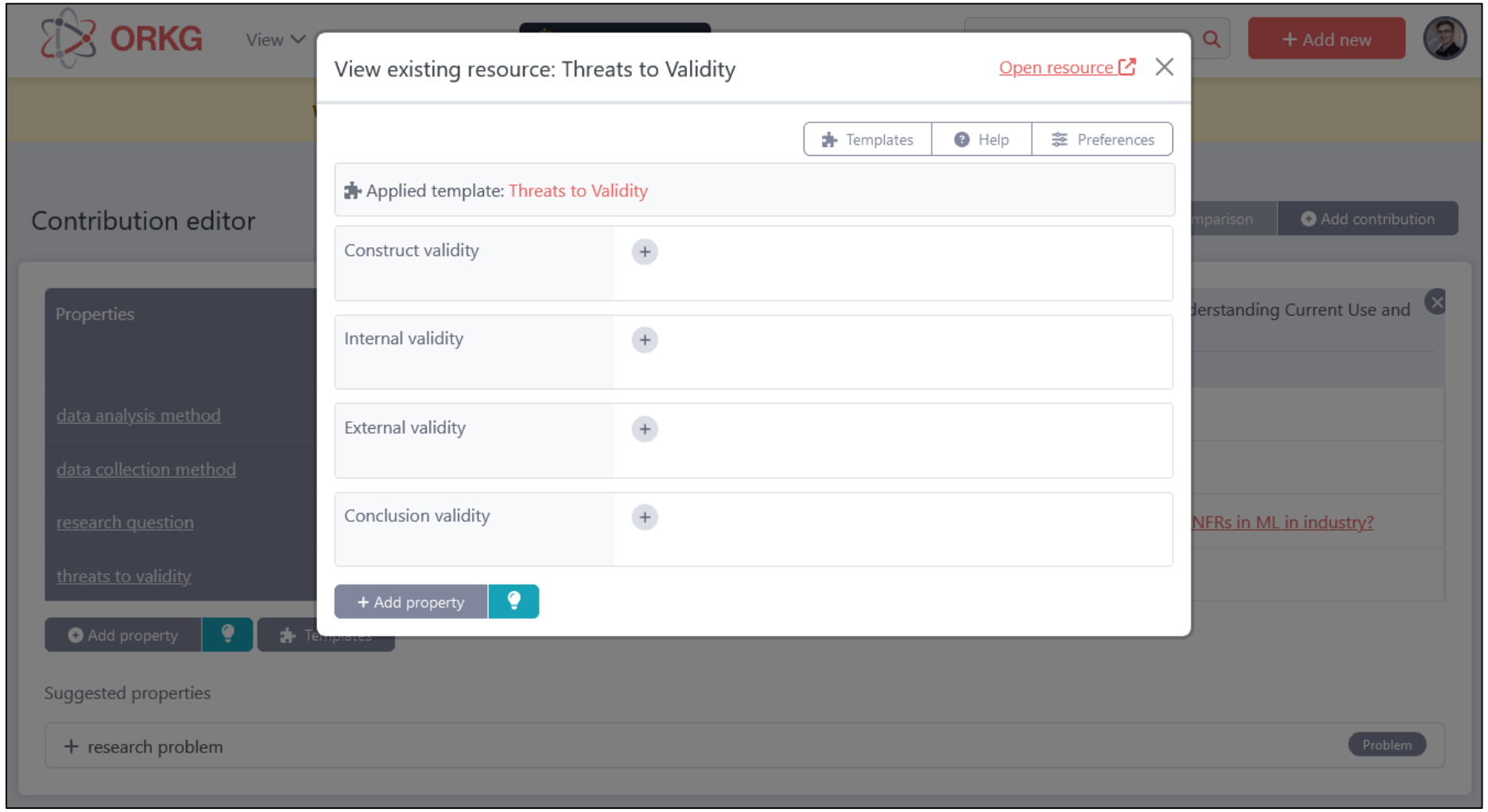
6. Describe a Paper: Enter the Research Question(s)



If you have more research questions, **repeat** the process.

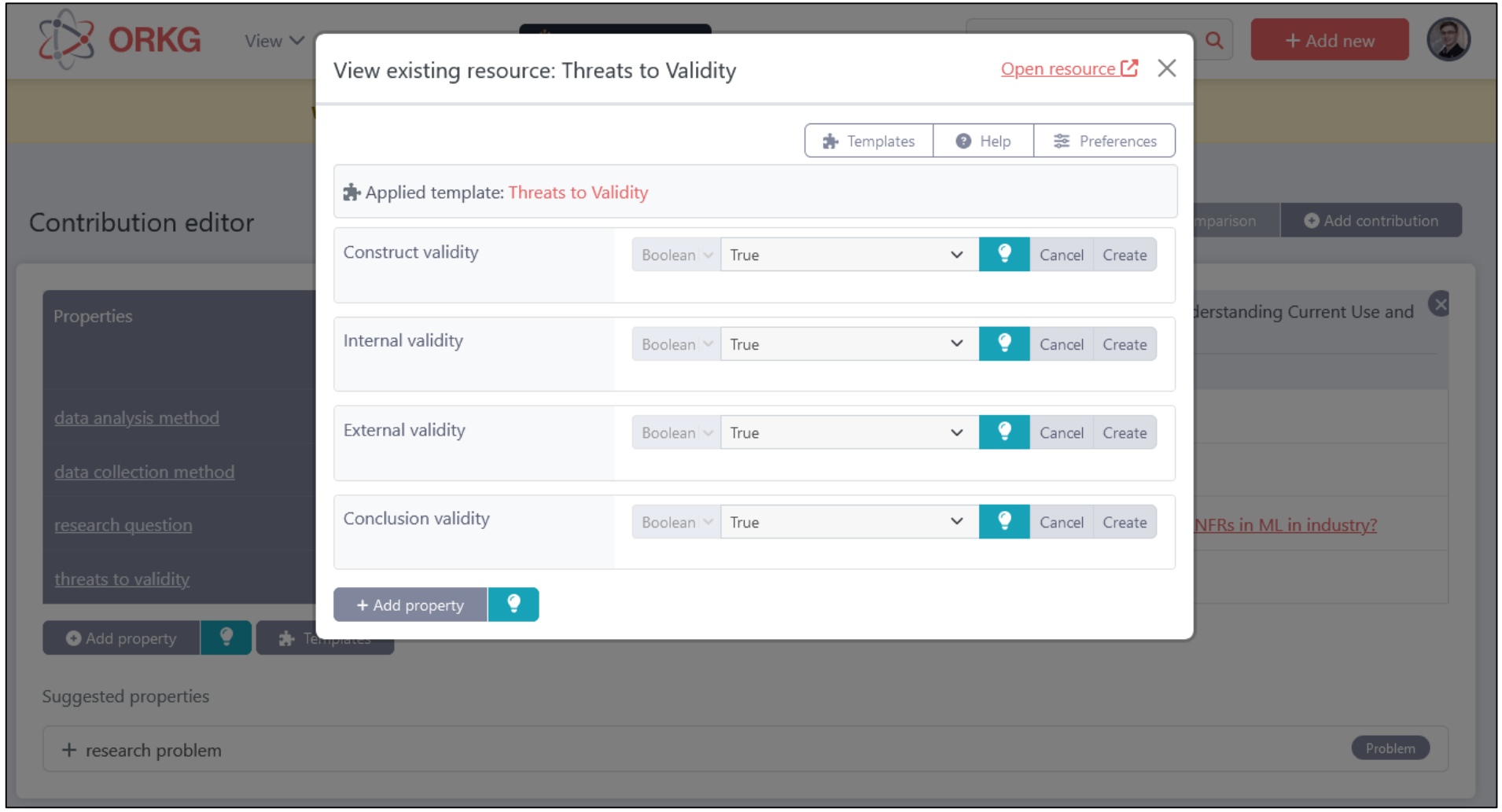
Otherwise, continue with the next property.

6. Describe a Paper: Enter the Threats to Validity



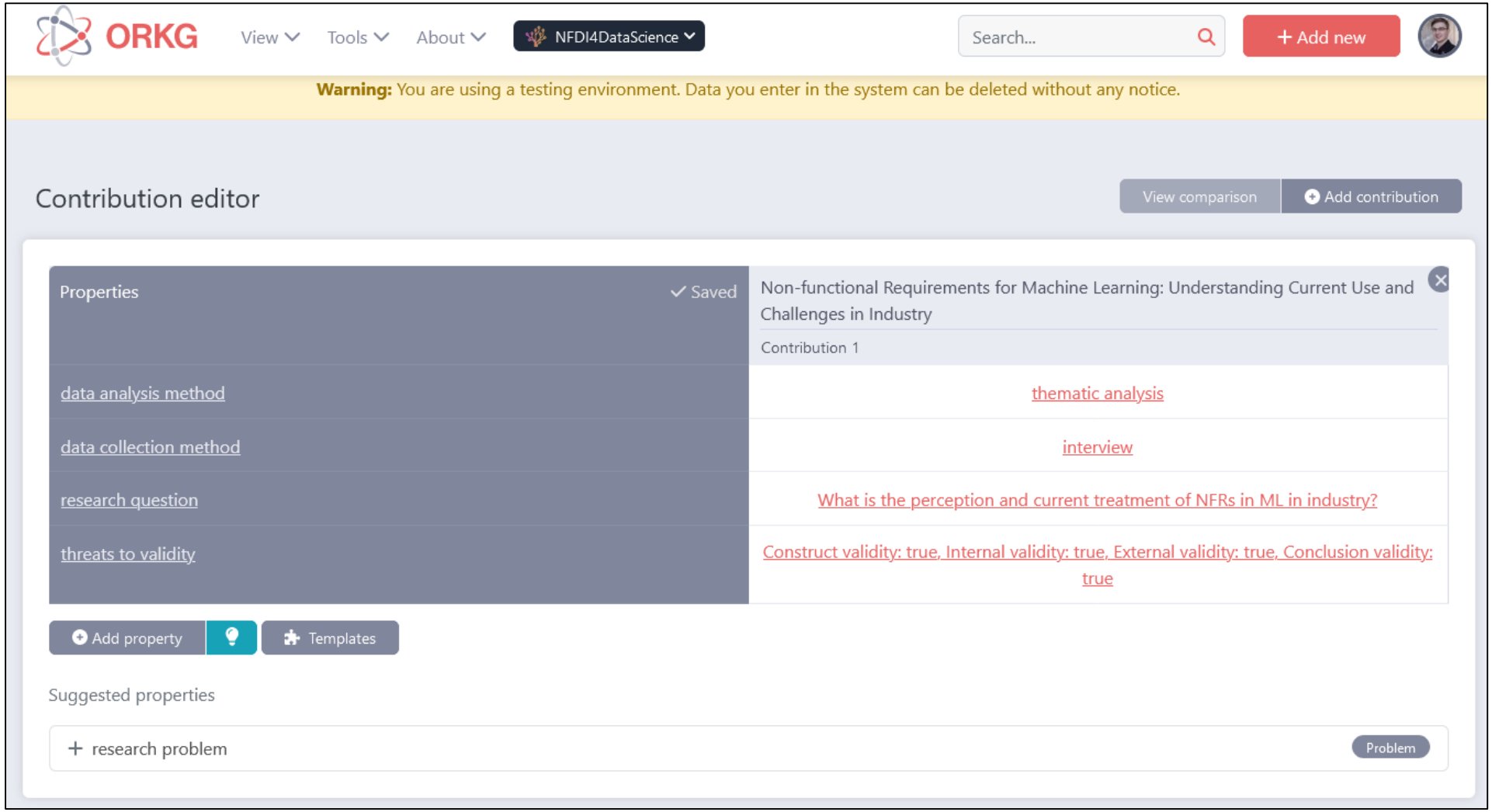
1. Click the ⊕
2. Enter construct validity
3. Click the ⊕
4. Enter internal validity
5. Click the ⊕
6. Enter external validity
7. Click the ⊕
8. Enter conclusion validity

6. Describe a Paper: Enter the Threats to Validity



1. Click “Create”
2. Click “Create”
3. Click “Create”
4. Click “Create”
5. Close dialog

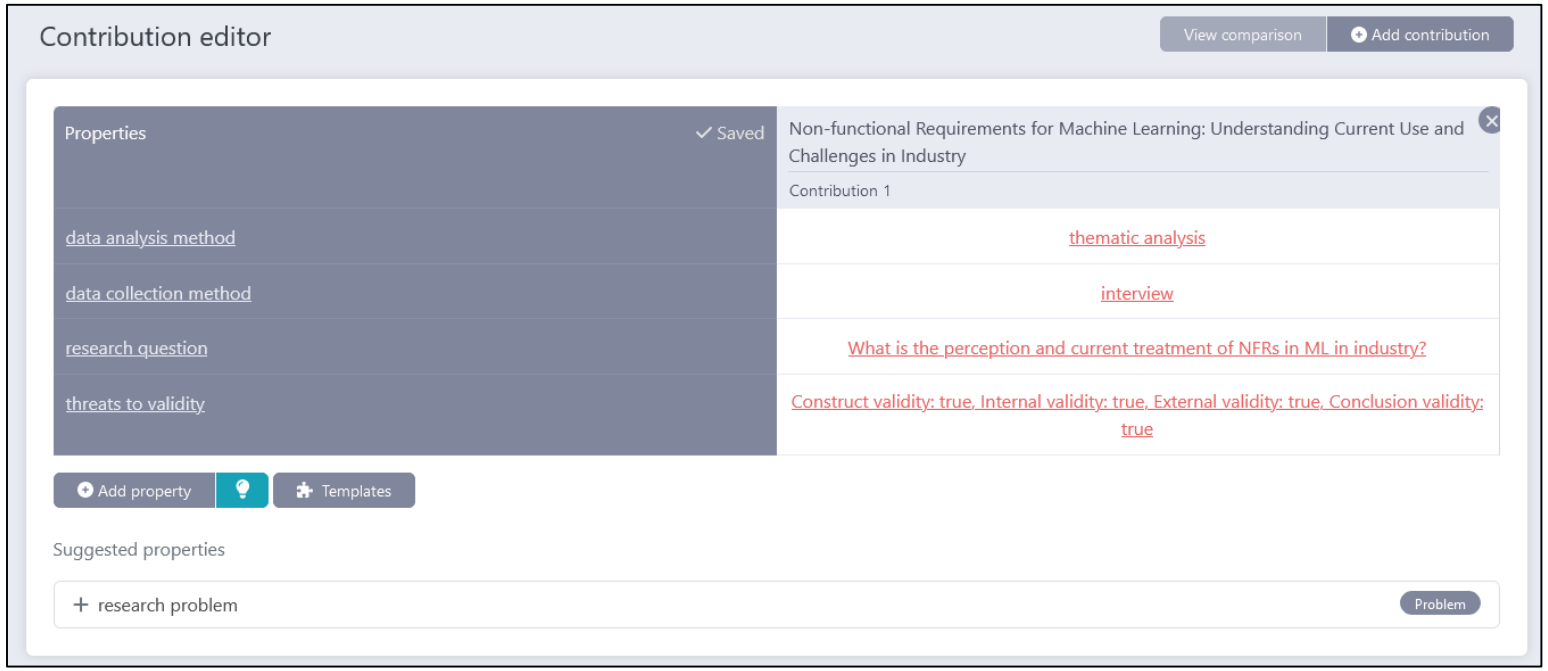
6. Describe a Paper: Enter the Threats to Validity



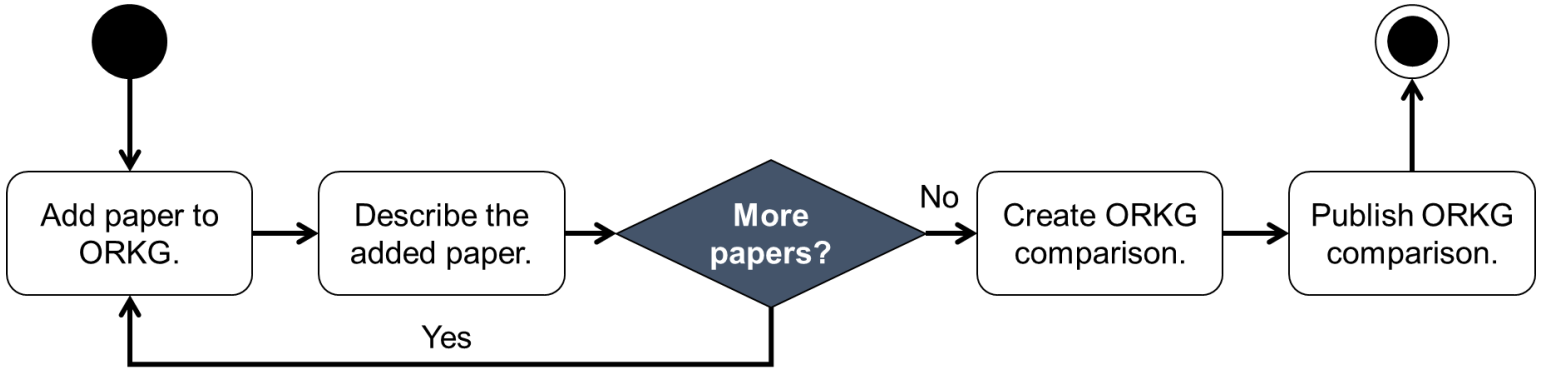
Now, we have described our first paper regarding its reported empirical research.

We can repeat the entire process for the next paper or work collaboratively!

7. Add the Next Paper

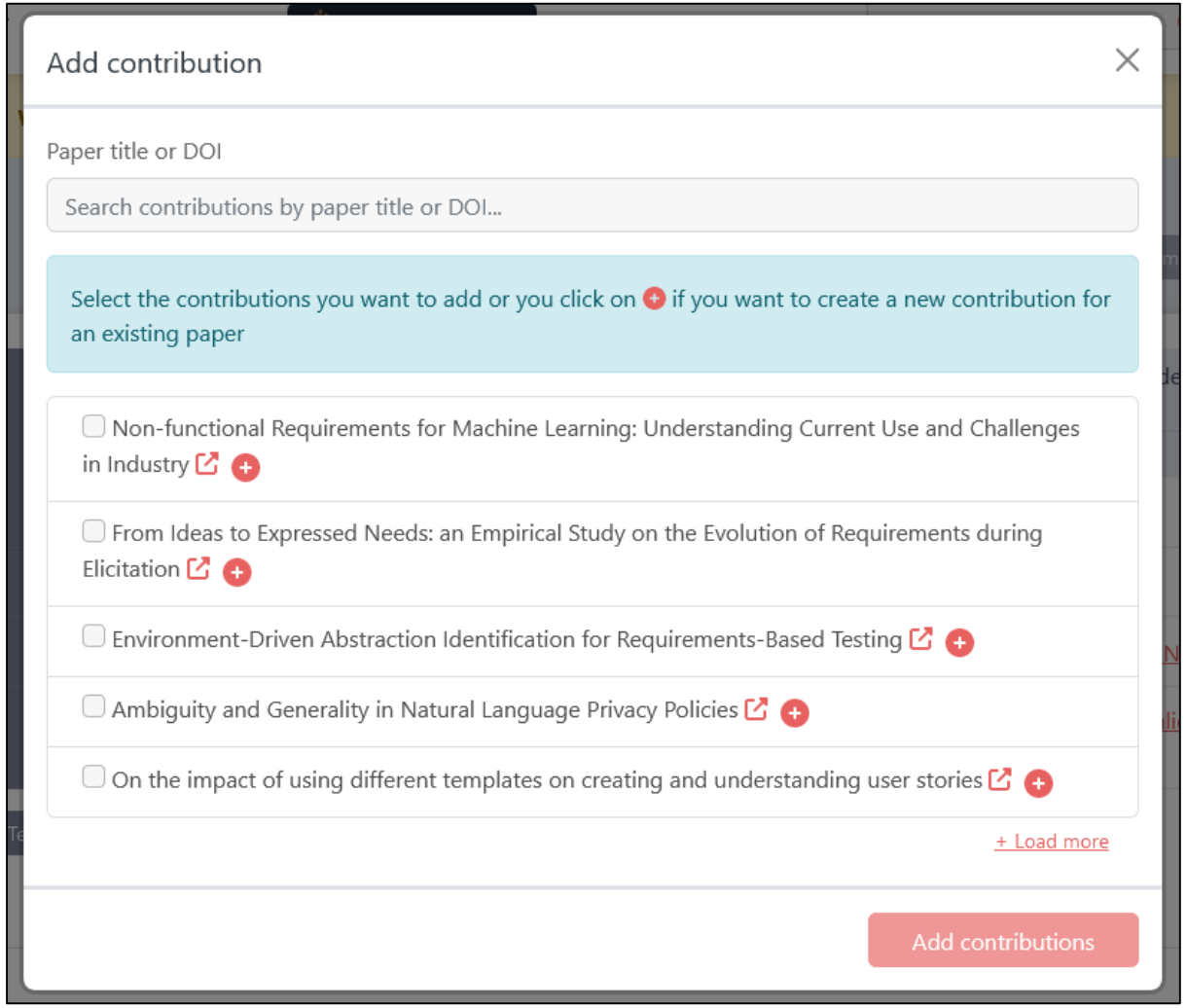


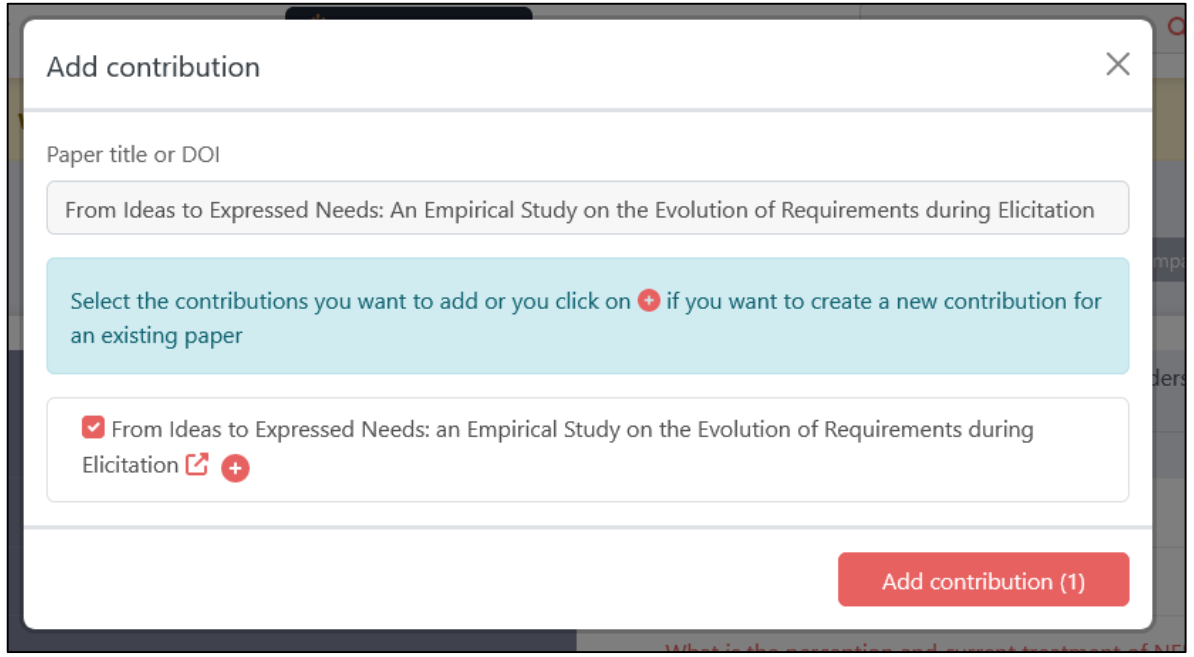
1. Click on “Add contribution”



Remark:
Now, we use a paper called “**From Ideas to Expressed Needs: an Empirical Study on the Evolution of Requirements during Elicitation**” already described with the ORKG template.

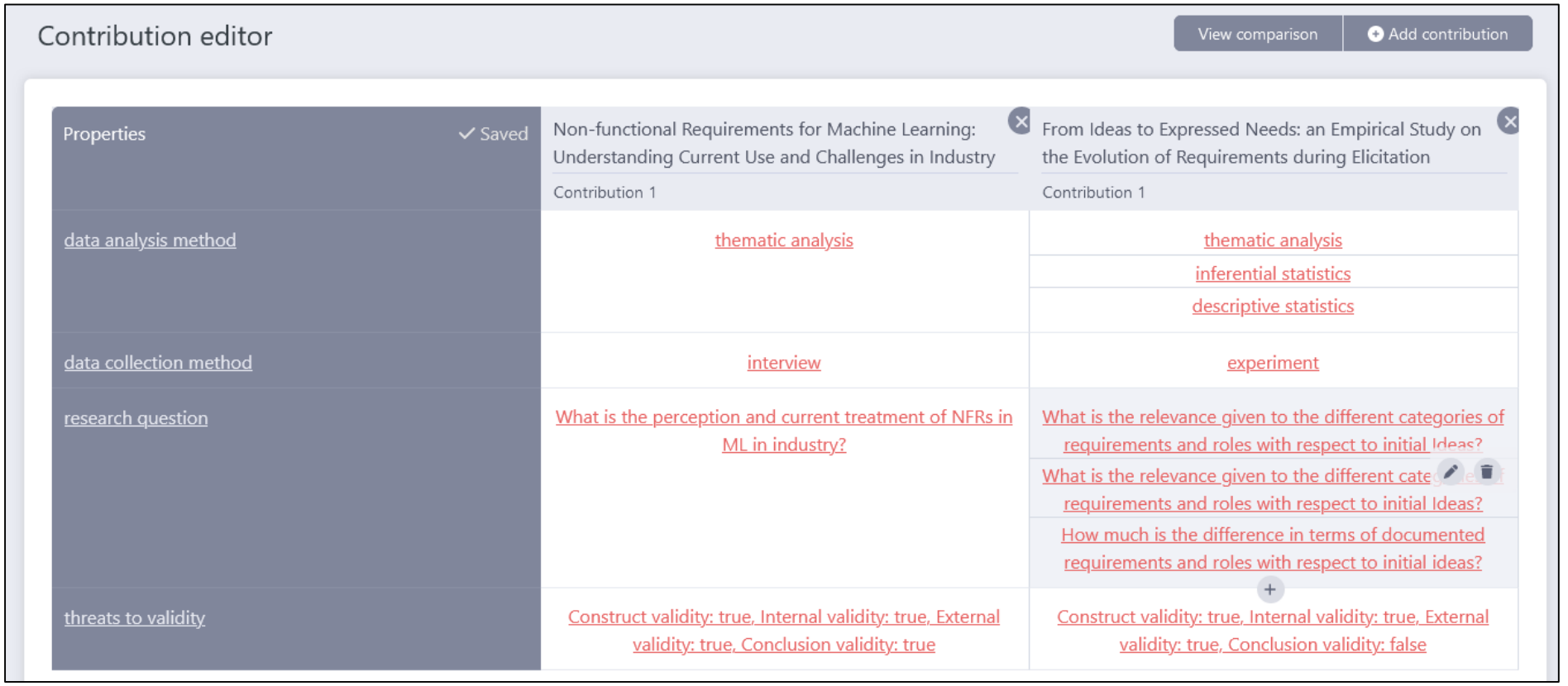
7. Add the Next Paper: Enter Paper Title





1. Enter paper title
2. Select the checkbox
3. Click “Add contribution (1)”

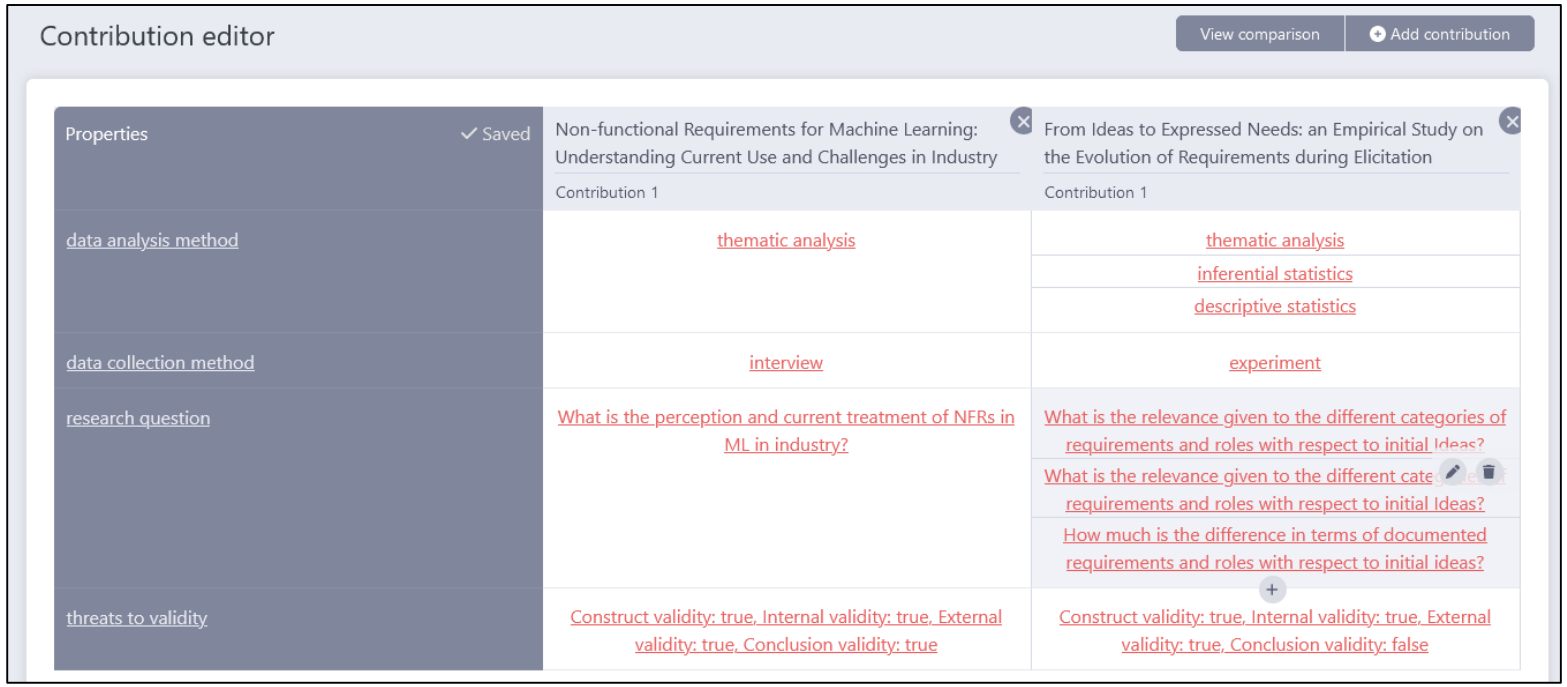
7. Add the Next Paper: Result



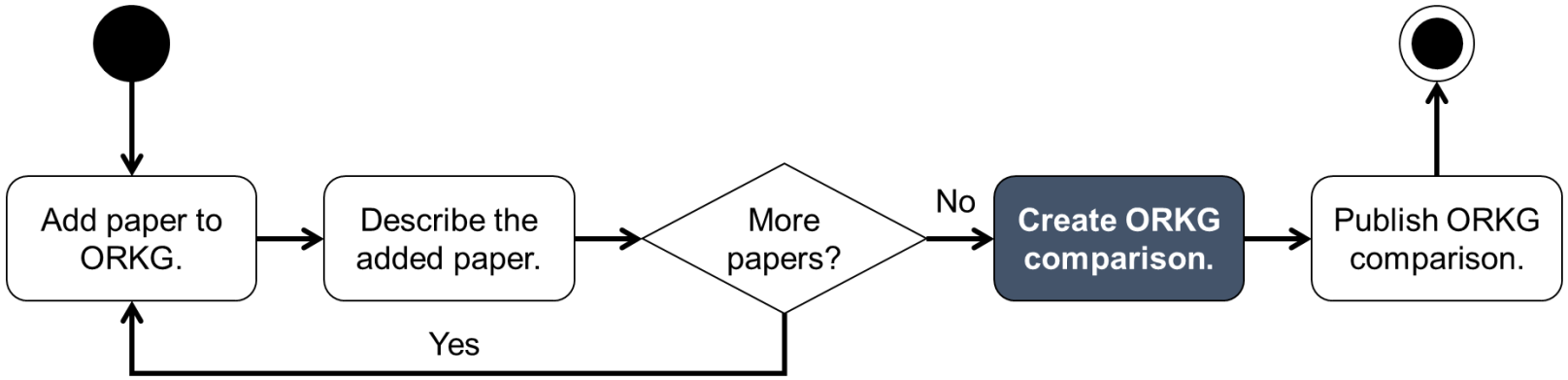
Now, we have two described papers.

If we want, we can add further ones or we can create an ORKG comparison.

8. Create ORKG Comparison



1. Click on “View comparison”



8. Create ORKG Comparison: Result and Improvements

Comparison | 2 contributions

Edit | + Add contribution | Visualize | Actions

Properties	Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry *Contribution 1 - 2021*	From Ideas to Expressed Needs: an Empirical Study on the Evolution of Requirements during Elicitation *Contribution 1 - 2021*
data analysis method	Data Analysis Method	Data Analysis Method
		Data Analysis Method
		Data Analysis Method
data analysis method/data analysis method/method*	thematic analysis	descriptive statistics
		inferential statistics
		thematic analysis
data collection method	Data Collection Method	Data Collection Method
data collection method/data collection method		
↳ method*	interview	experiment
↳ number of participants*	10	30
research question	Research Question	Research Question
		Research Question
		Research Question
research question/research question		
↳ hidden in text*	✗	✗
		✗
		✗
↳ highlighted in text*	✓	✓
		✓
		✓
↳ question*	What is the perception and current treatment of NFRs in ML in industry?	How much is the difference in terms of documented requirements and roles with respect to initial ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
threats to validity	Threats to Validity	Threats to Validity
threats to validity/threats to validity		
↳ conclusion validity*	✓	✗
↳ construct validity*	✓	✓
↳ external validity*	✓	✓
↳ internal validity*	✓	✓

Now, we **created our ORKG comparison**.

Options before publishing the ORKG comparison:

1. Edit the ORKG comparison by ordering the rows

   Edit → Drag & Drop the property cells as required

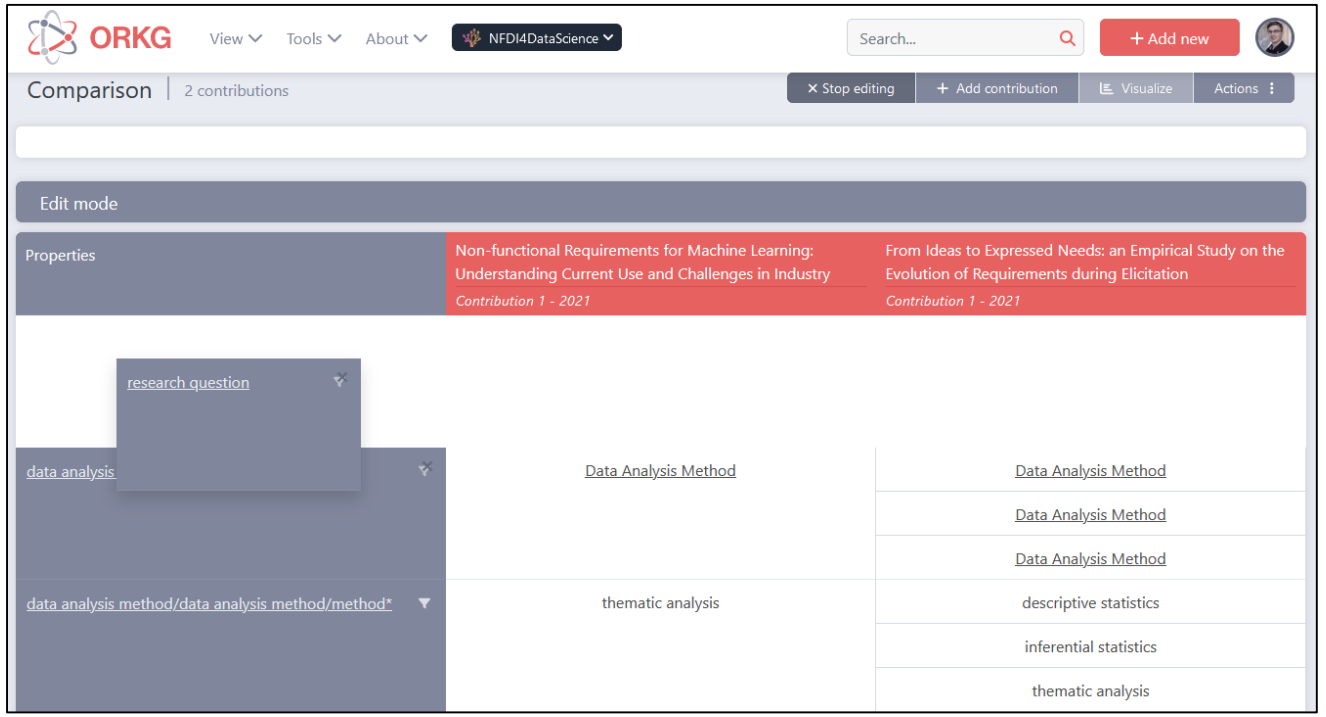
2. Select properties we want to show.

   Actions → Select properties → Disable checkboxes of properties to hide

3. Save ORKG comparison as a draft for later

   Actions → Save as draft → Enter title → Save → Draft in “My account”

9. Improve ORKG Comparison: Order Rows



1. Click on “Edit”
2. Click on the grey property cell you want to move
3. Drag & Drop the property cell where you want

9. Improve ORKG Comparison: Order Rows – Result

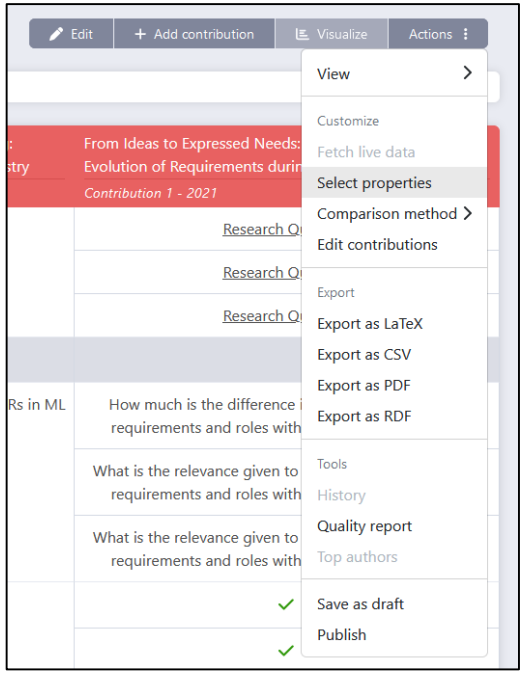
Data Analysis

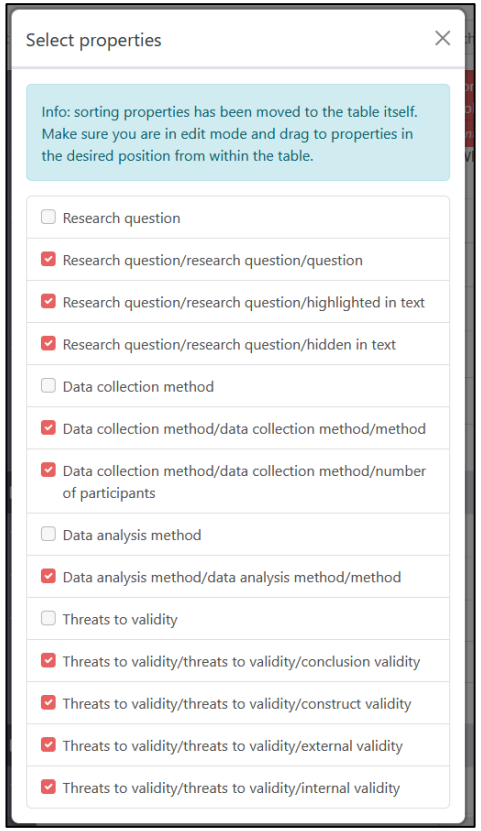
Data Collection

Research Question

Threats to Validity

9. Improve ORKG Comparison: Select Properties





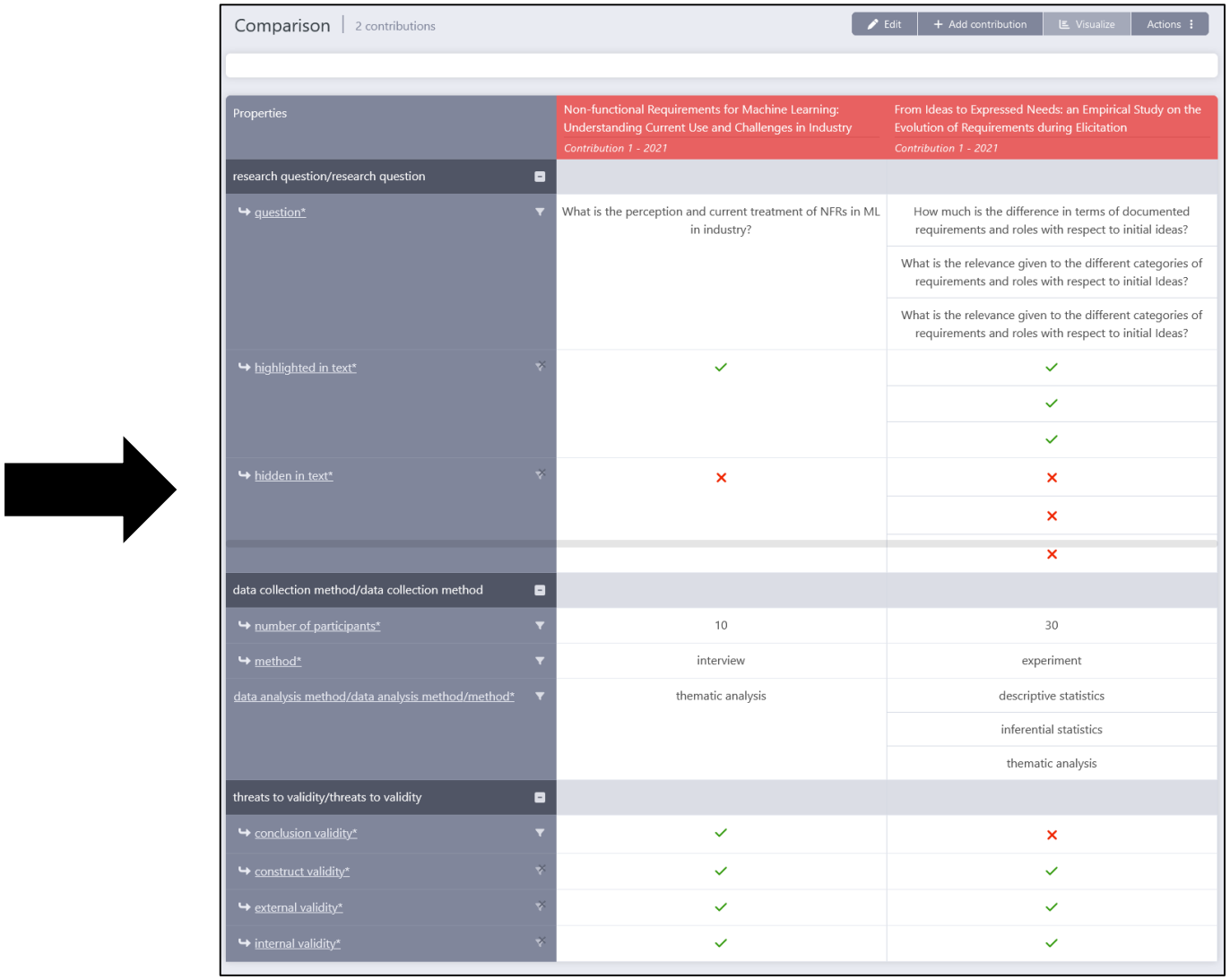
1. Click on “Actions”
2. Click on “Select properties”
3. Disable all checkboxes of properties you want to hide
4. Close dialog

9. Improve ORKG Comparison: Select Properties – Result

Comparison | 2 contributions

Edit | + Add contribution | Visualize | Actions

Properties	Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry *Contribution 1 - 2021*	From Ideas to Expressed Needs: an Empirical Study on the Evolution of Requirements during Elicitation *Contribution 1 - 2021*
research question	Research Question	Research Question
		Research Question
		Research Question
research question/research question		
question*	What is the perception and current treatment of NFRs in ML in industry?	How much is the difference in terms of documented requirements and roles with respect to initial ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
highlighted in text*	✓	✓
		✓
		✓
hidden in text*	✗	✗
		✗
		✗
data collection method	Data Collection Method	Data Collection Method
data collection method/data collection method		
method*	interview	experiment
number of participants*	10	30
data analysis method	Data Analysis Method	Data Analysis Method
		Data Analysis Method
		Data Analysis Method
data analysis method/data analysis method/method*	thematic analysis	descriptive statistics
		inferential statistics
		thematic analysis
threats to validity	Threats to Validity	Threats to Validity
threats to validity/threats to validity		
conclusion validity*	✓	✗
construct validity*	✓	✓
external validity*	✓	✓
internal validity*	✓	✓

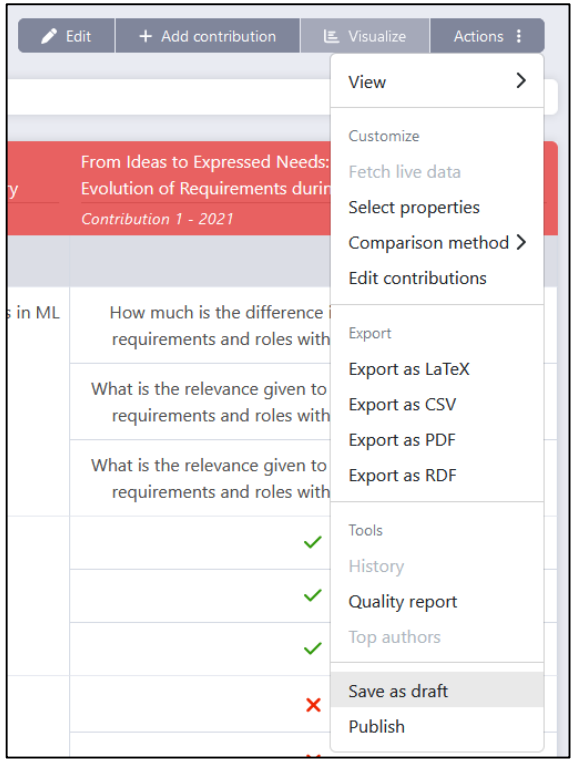


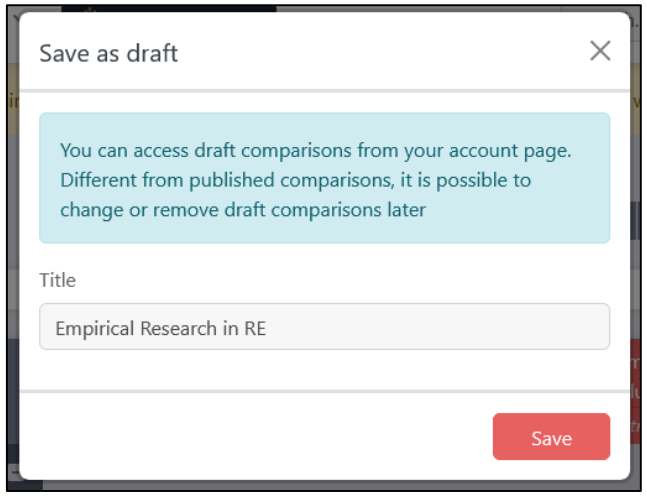
Comparison | 2 contributions

Edit | + Add contribution | Visualize | Actions

Properties	Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry *Contribution 1 - 2021*	From Ideas to Expressed Needs: an Empirical Study on the Evolution of Requirements during Elicitation *Contribution 1 - 2021*
research question/research question		
question*	What is the perception and current treatment of NFRs in ML in industry?	How much is the difference in terms of documented requirements and roles with respect to initial ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
		What is the relevance given to the different categories of requirements and roles with respect to initial Ideas?
highlighted in text*	✓	✓
		✓
		✓
hidden in text*	✗	✗
		✗
		✗
data collection method/data collection method		
number of participants*	10	30
method*	interview	experiment
data analysis method/data analysis method/method*	thematic analysis	descriptive statistics
		inferential statistics
		thematic analysis
threats to validity/threats to validity		
conclusion validity*	✓	✗
construct validity*	✓	✓
external validity*	✓	✓
internal validity*	✓	✓

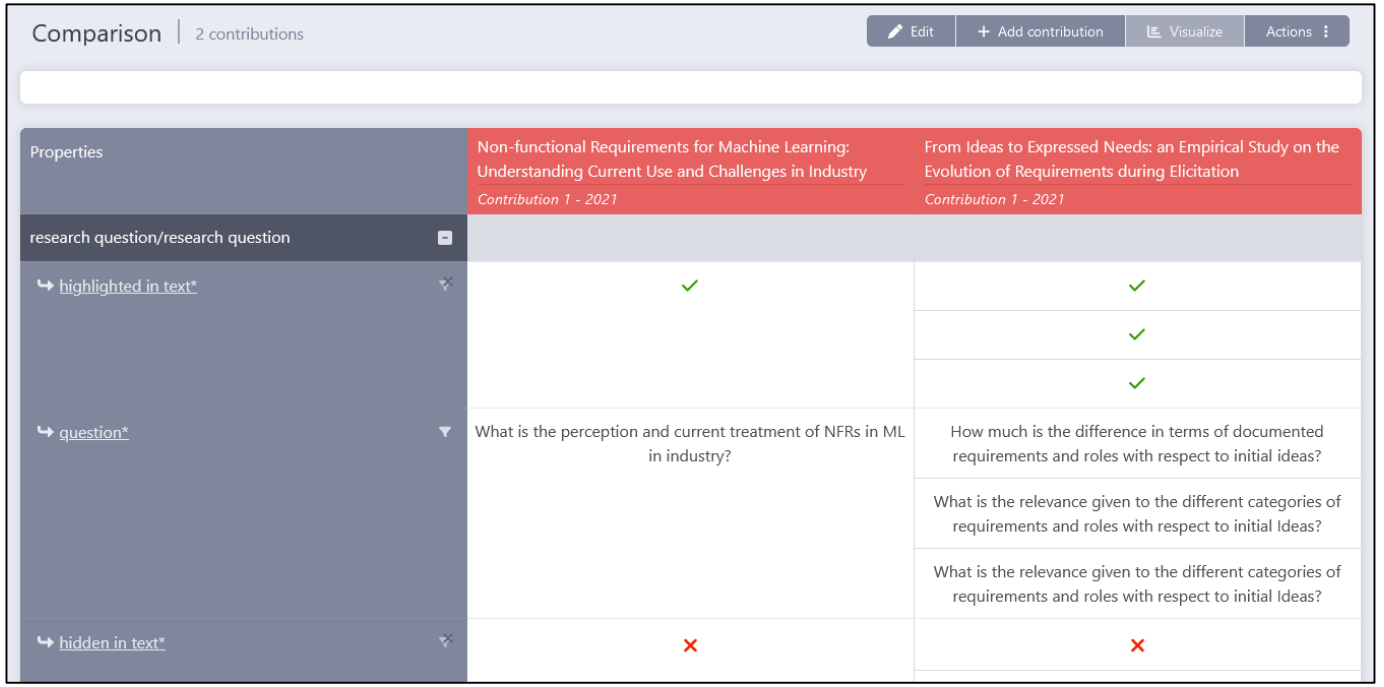
9. Improve ORKG Comparison: Save as Draft

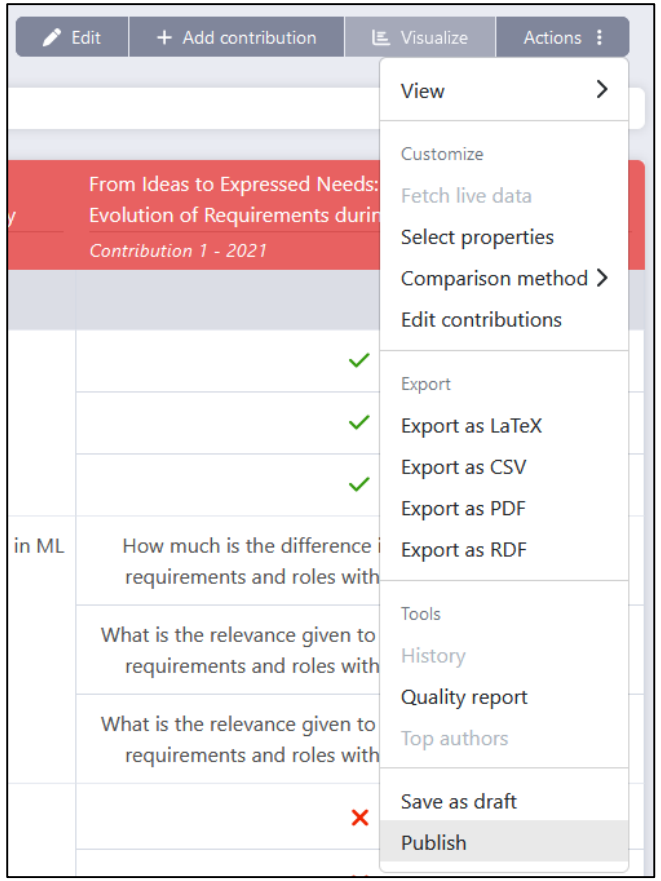


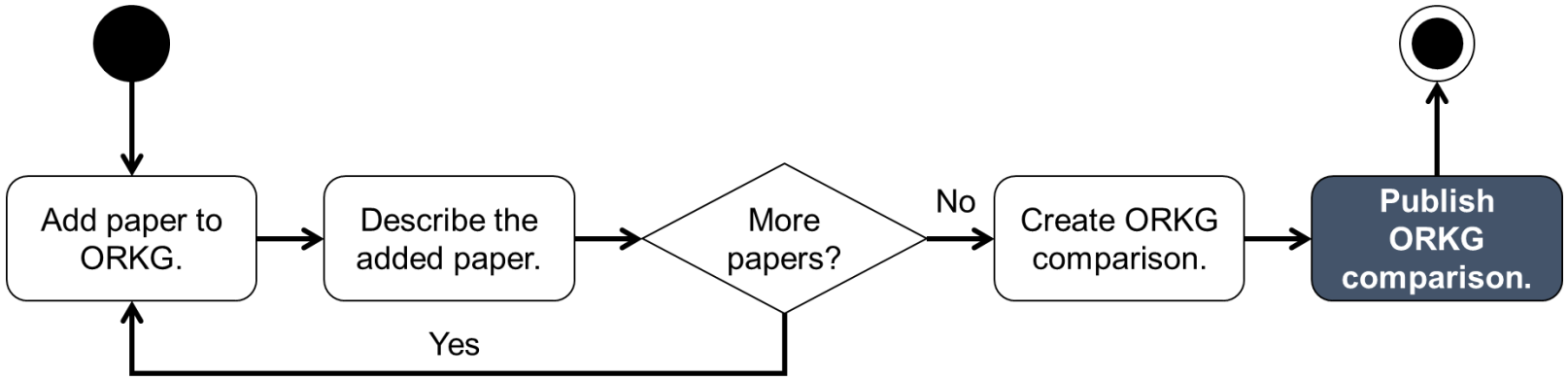


1. Click on “Actions”
2. Click on “Save as draft”
3. Enter a title and click on “Save”
4. The draft is saved in your account

10. Publish ORKG Comparison

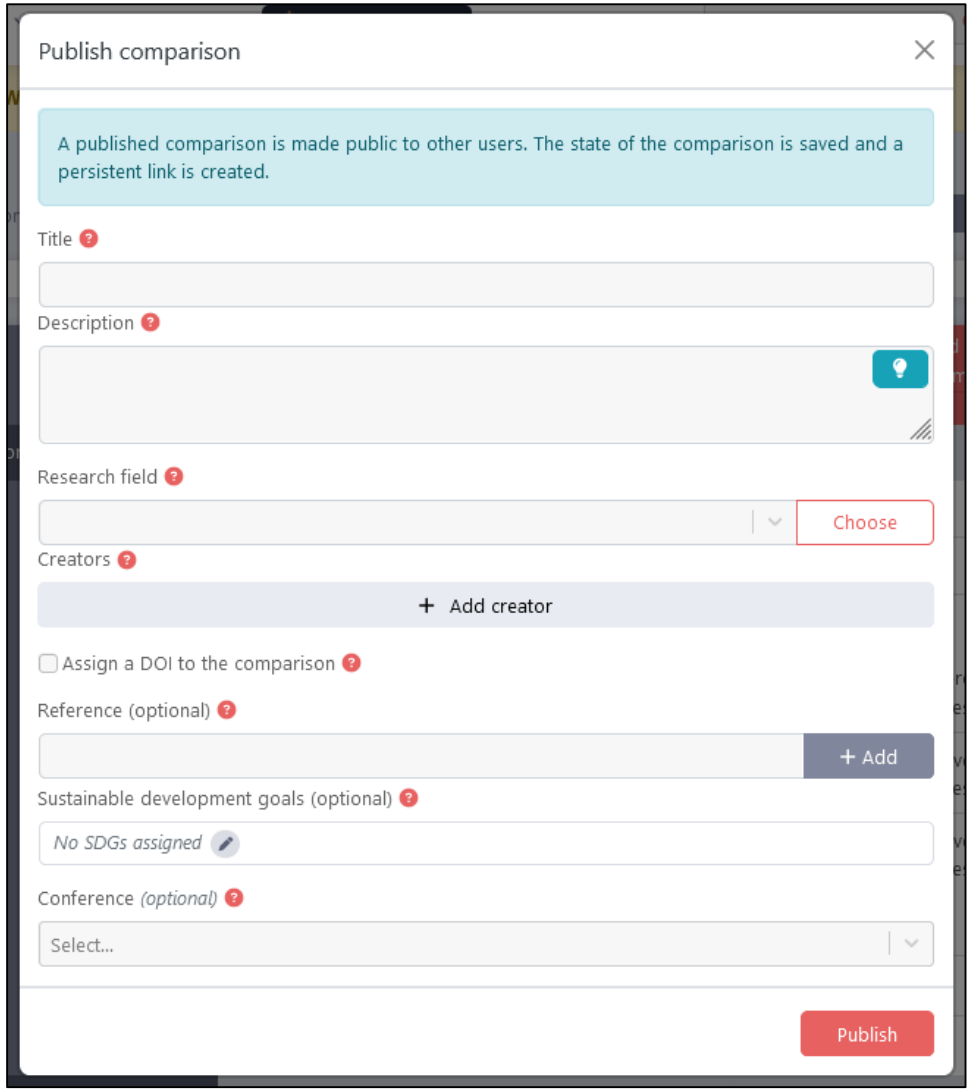






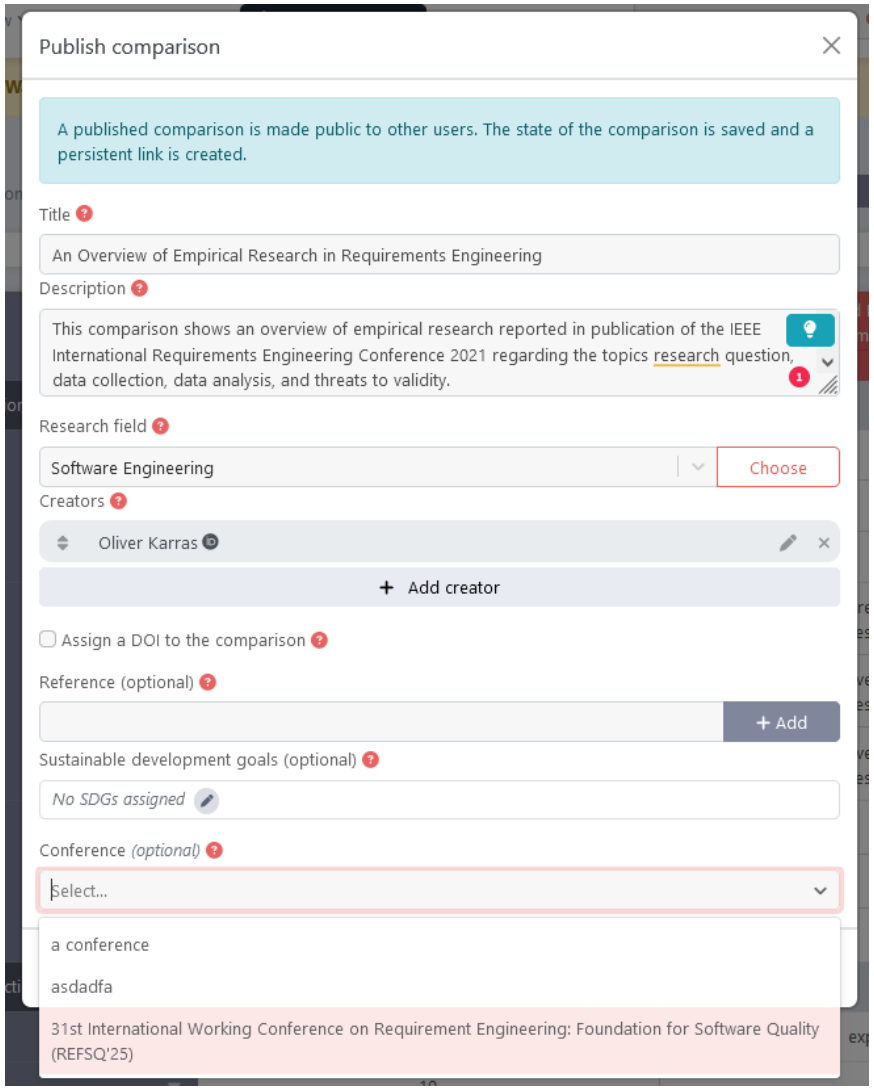
1. Click on “Actions”
2. Click on “Publish”

10. Publish ORKG Comparison

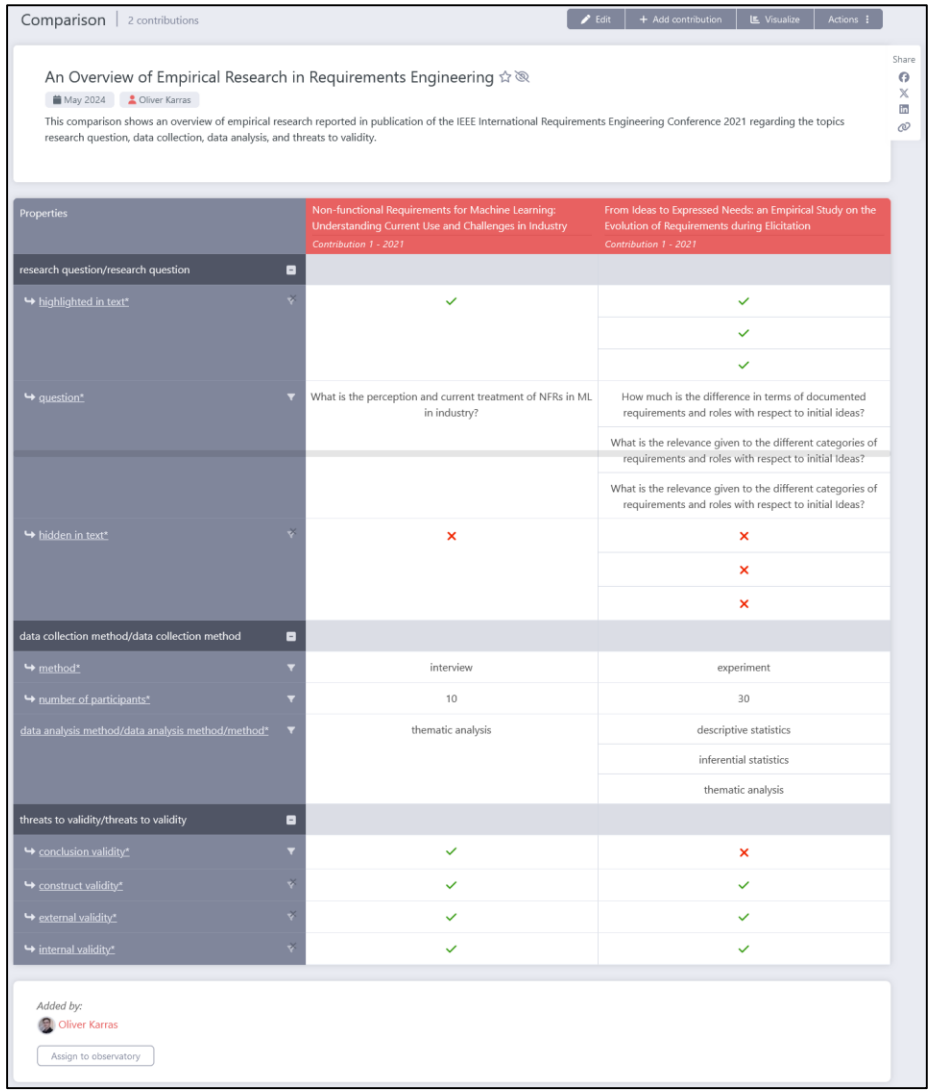


1. Fill out the dialog
2. Click on “Publish”

Remark:
You can add
- A **DOI** for citation (can also be done later)
- Additional **References**
- Related **Sustainable development goals**
- ORKG partner **Conferences**, such as **REFSQ’25**, that award the **Best ORKG Comparison Award**



10. Publish ORKG Comparison: Result



Now, we **published our ORKG comparison**, a stable version that can be maintained, extended, updated, and published as new versions.

Options after publishing the ORKG comparison:

1. Add visualizations
2. Add DOI later and export citation
3. Use the quality report to get feedback from other researchers
4. Fetch live data for a new draft or published version
5. Fetch data for later analysis with different interfaces

Remark:
These options are only demonstrated live in the tutorial. If you need help, do not hesitate to contact the ORKG team, especially **Oliver Karras** (oliver.karras@tib.eu).

Summary, Reflection, and Closing

Oliver Karras, Alessio Ferrari, Davide Fucci, and Davide Dell'Anna

oliver.karras@tib.eu, alessio.ferrari@isti.cnr.it, davide.fucci@bth.se, d.dellanna@uu.nl

32nd IEEE International Requirements Engineering 2024 Conference – Exploring New Horizons: Expanding the Frontiers of Requirements Engineering

June 24th, 2024, Reykjavik, Iceland

Summary

Mining User Requirements from Application Store Reviews Using Frame Semantics

Nishant Jha and Anas Mahmoud(✉)

The Division of Computer Science and Engineering, Louisiana State University, Baton Rouge, LA 70803, USA
njha1@lsu.edu, mahmoud@csc.lsu.edu

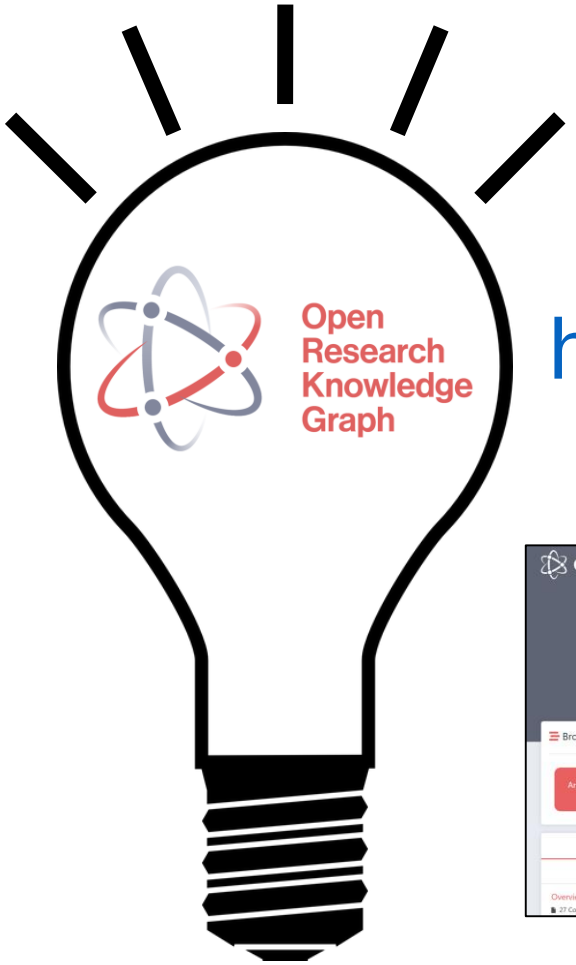
Abstract. *Context and motivation:* Research on mining user reviews in mobile application (app) stores has noticeably advanced in the past few years. The majority of the proposed techniques rely on classifying the textual description of user reviews into different categories of technically informative user requirements and uninformative feedback. *Question/Problem:* Relying on the textual attributes of reviews often produces high dimensional models. This increases the complexity of the classifier and can lead to overfitting problems. *Principal ideas/results:* We propose a novel semantic approach for app review classification. The proposed approach is based on the notion of semantic role labeling, or characterizing the lexical meaning of text in terms of semantic frames. Semantic frames help to generalize from text (individual words) to more abstract scenarios (contexts). This reduces the dimensionality of the data and enhances the predictive capabilities of the classifier. Three datasets of user reviews are used to conduct our experimental analysis. Results show that semantic frames can be used to generate lower dimensional and more accurate models in comparison to text classification methods. *Contribution:* A novel semantic approach for extracting user requirements from app reviews. The proposed approach enables a more efficient classification process and reduces the chance of overfitting.

Keywords: Requirements elicitation · Application stores · Classification

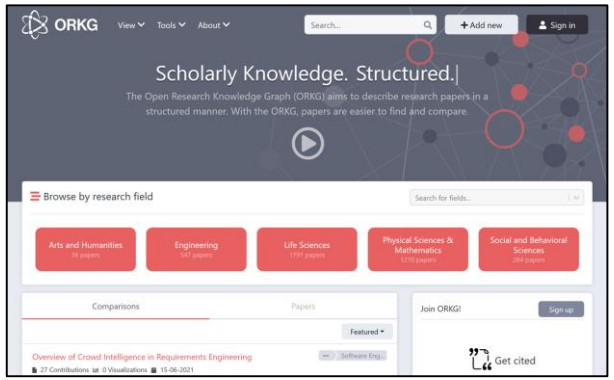
1 Introduction

Mobile application markets, or app stores (e.g., Google Play and Apple App Store), represent a unique model of service-oriented business. Such platforms have created an unprecedented opportunity for app developers to directly monitor the opinions of a large population of end-users of their software [25]. Through app stores feedback services, app users can directly share their experience in the form of textual reviews and meta-data (e.g., star ratings). Analyzing large datasets of app store reviews has revealed that they contain substantial amounts of up-to-date technical information. Such information can be leveraged by app developers to help them maintain and sustain their apps in a highly-competitive

P. Grünbacher and A. Perini (Eds.): REFSQ 2017, LNCS 10153, pp. 273–287, 2017.
DOI: 10.1007/978-3-319-54045-0_20



https://orkg.org



Let's bring scholarly communication and open science in Requirements Engineering to the 21st century!

Recap of the Tutorial

Session	Time	Table of Content	Style	Speaker
Theoretical	09:00 - 09:25	1. Welcome (5 min)	Presentation	All organizers
		2. Introduction to open science in RE (10 min)	Presentation	Alessio Ferrari
		3. Introduction to the ORKG (10 min)	Presentation	Oliver Karras
Practical	09:25 - 10:15	4. Create a FAIR-annotated publication for the ORKG (50 min)	Exercise	Oliver Karras
		4.1 Set up an Overleaf project for an exemplary publication	Sub-exercise	All organizers
		4.2 Use the LaTex package SciKGTeX to annotate the publication	Sub-exercise	All organizers
		4.3 Generate PDF with embedded FAIR scientific information	Sub-exercise	All organizers
		4.4 Optional: Upload the FAIR-annotated publication to the ORKG	Sub-exercise	All organizers
Break	10:15 - 10:45	Coffee break		
Practical	10:45 - 11:45	5. Use the ORKG based on a RE use case (60 min)	Exercise	Oliver Karras
		5.1 Add an exemplary publication to the ORKG	Sub-exercise	All organizers
		5.2 Describe the scientific information of the publication in the ORKG	Sub-exercise	All organizers
		5.3 Create an ORKG comparison of the publications added by participants	Sub-exercise	All organizers
		5.4 Publish the created ORKG comparison as a citable digital artifact	Sub-exercise	All organizers
		5.5 Optional: Create visualizations for the created ORKG comparison	Sub-exercise	All organizers
		5.6 Optional: Retrieve the information with the SPARQL endpoint	Sub-exercise	All organizers
Feedback	11:45 - 12:15	6. Reflection of the tutorial with the participants (25 min)	Discussion	All organizers
		7. Farewell and closing (5 min)	Presentation	All organizers

Reflection and Feedback

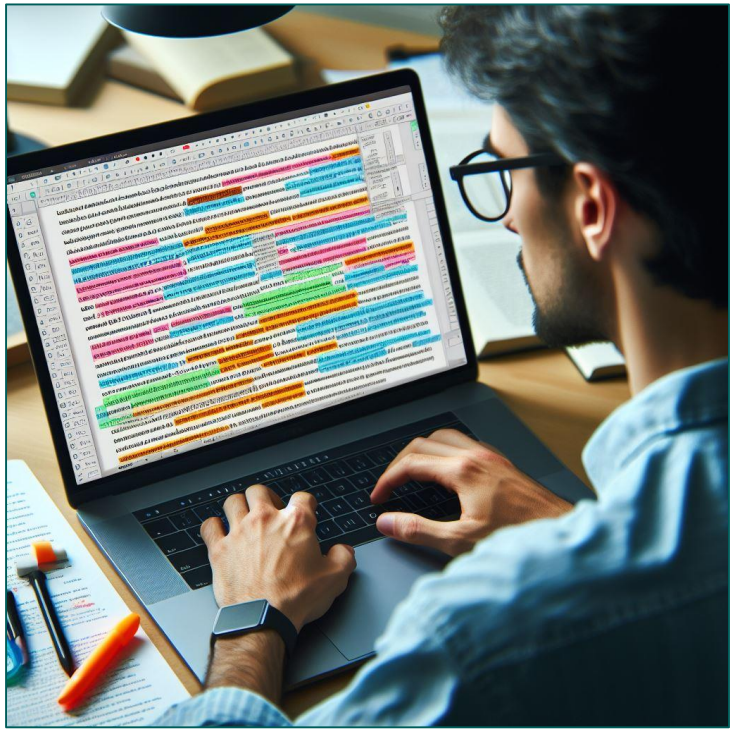
Your thoughts, please!

Teaser: REFSQ'25 Open Science Competition

Challenge 1:

Annotate your REFSQ'25 paper with SciKGTeX.

The accepted paper, best annotated with SciKGTeX, will be awarded the **Best ORKG Annotation Award** (prize:100€).



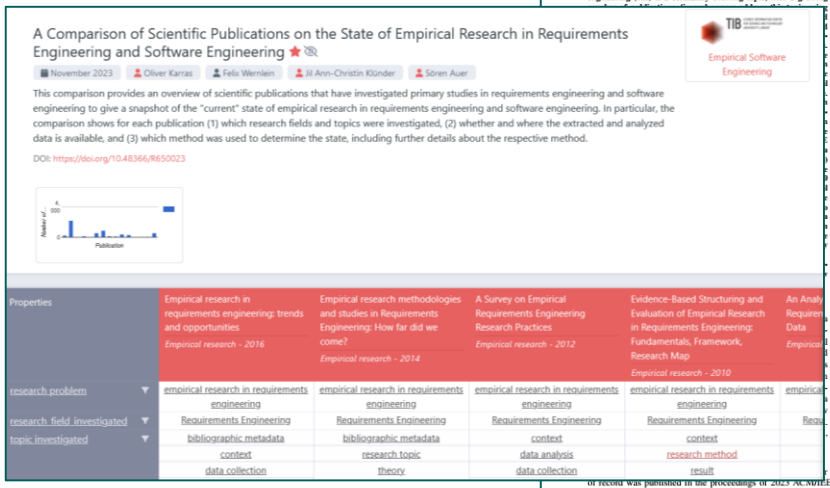
SciKGTeX

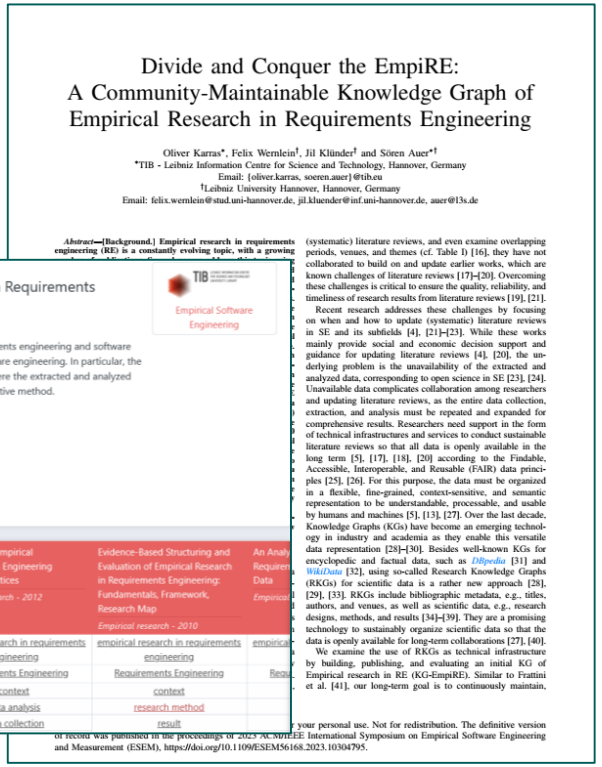
Challenge 2:

Enrich your RESFQ'25 paper with an ORKG comparison.

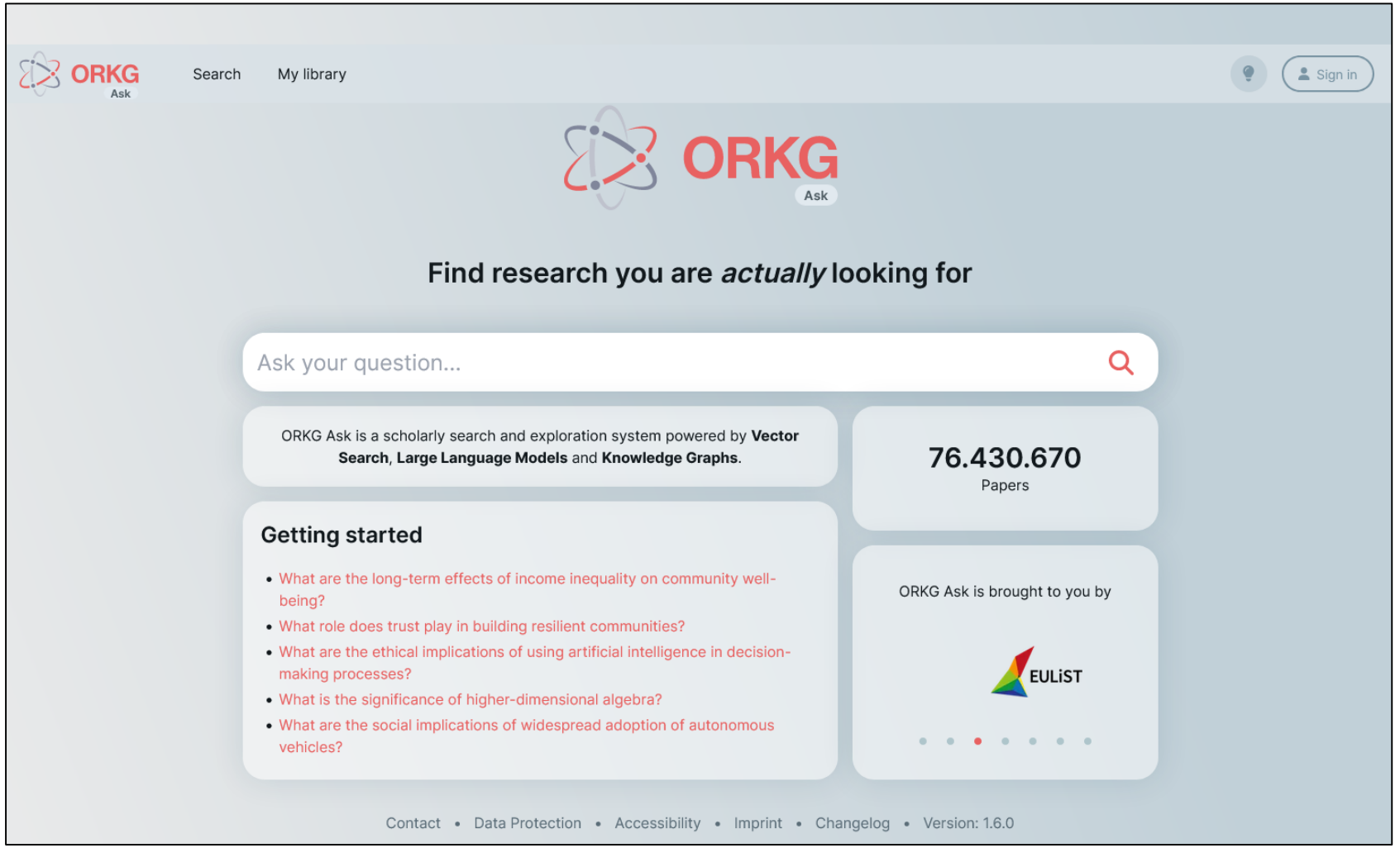
The accepted paper, enriched with the best ORKG comparison, will be awarded the **Best ORKG Comparison Award** (prize: 200€).

Open Research Knowledge Graph


A Comparison of Scientific Publications on the State of Empirical Research in Requirements Engineering and Software Engineering


Divide and Conquer the EmpiRE:
A Community-Maintainable Knowledge Graph of Empirical Research in Requirements Engineering

Looking to the Future: ORKG Ask



The ORKG Ask is a scientific **search and exploration system** that helps researchers **find** the **research articles** they are really looking for.

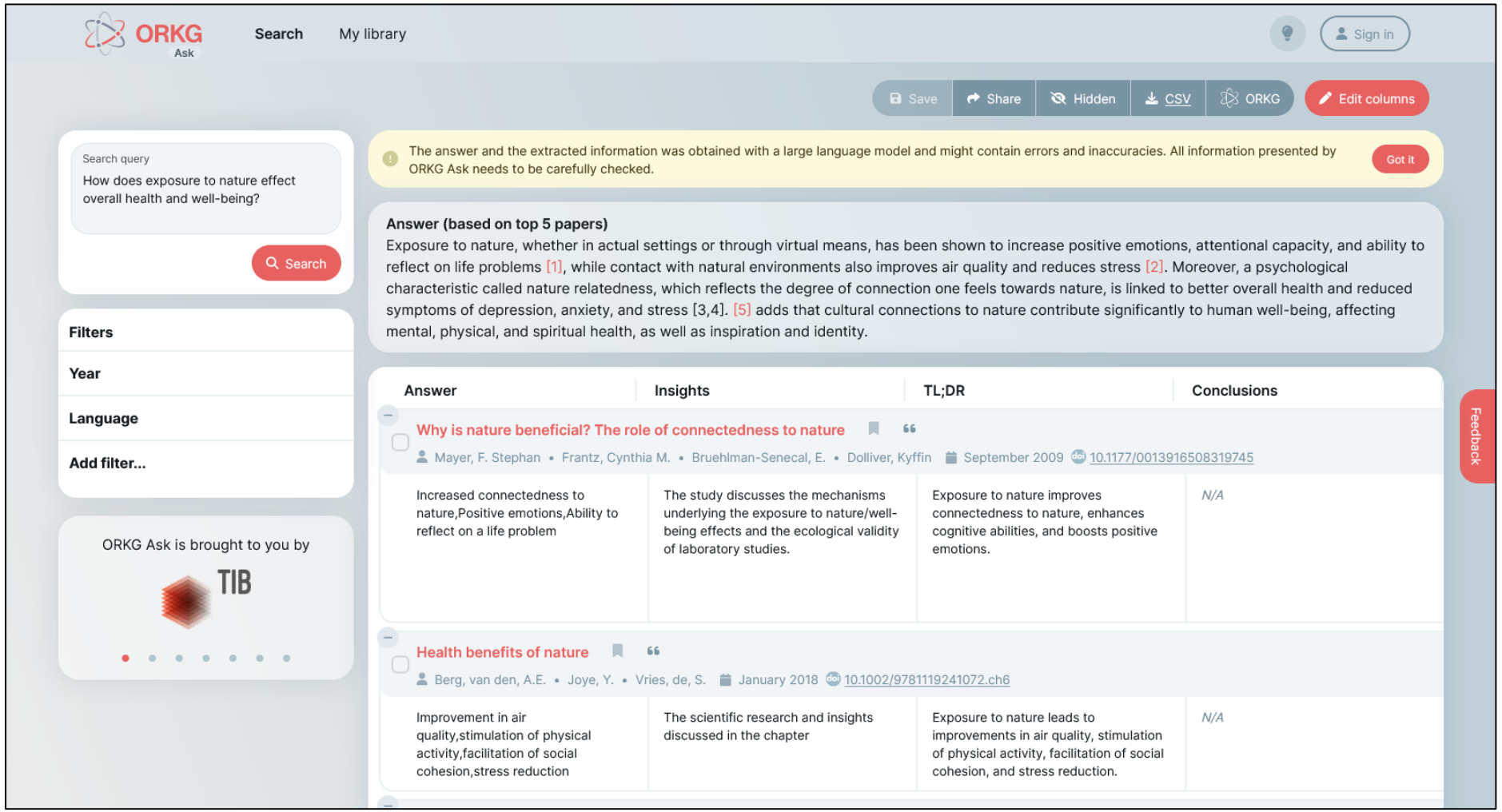
Ask a **question**.

Get an **answer** with **references**.

https://ask.orkg.org/

(Released: May16th, 2024)

Looking to the Future: ORKG Ask



1. Enter a natural language **question** in the **UI**
2. **Semantic search** finds the most relevant **papers**
3. **LLMs** extract the required information and create the **answer**
4. **KGs** are used to **support** content extraction, synthesis, and enhancement